Survey on Security Challenge for Data forwarding in Cloud

Tushar A. Rane[1], Shrishail T. Patil[2], Anita H. Khade[3]

[1]Department of IT, Pune Institute of
Computer Technology, Pune, India
ranetushar@yahoo.com

[2]Department of Computer, Vishwakarma Institute
of Technology, Pune, India
stpatil77@gmail.com

[3]Department of IT, Pune Institute of
Computer Technology, Pune, India
anita_khade@yahoo.co.in

Abstract: ***Cloud Computing emerged as the next-generation architecture of IT Enterprise. As high speed network or Internet access become available to users, so user can easily use many services through Internet from anywhere at any time. Cloud storage system as we called third party provides long term storage service over the internet. It migrate all application software as well as databases to the centralized data centers, which creates challenges in management of the data and services. This paradigm causes many new security challenges during data storage and data forwarding. Storing and movement of data in cloud system make a serious problem about data confidentiality. General encryption scheme protect data confidentiality and also faces some limitations during functioning of the storage system as well as data forwarding. For constructing secure cloud storage with safe data forwarding, here we use proxy cryptosystem and along with this propose conditional proxy re-encryption. The conditional proxy re-encryption used for secure encryption and secure data forwarding.***

Keywords: Cloud Computing, Proxy Re-Encryption, Conditional Proxy Re-encryption, Security.

1. Introduction

Now days, more and more users store their important data in cloud. To ensure the security of the remotely stored data, users need to encrypt important data. From the point of data security which has always been important aspect of quality of service, cloud computing cause's new challenging security threats.

Firstly, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted due to the users' loss control of data under cloud computing. Therefore, correct data storage in the cloud should be verified without any knowledge of the whole data. By considering the various types of data for each user stored in the cloud and the requirement of long term assurance of their data safety, the problem arises as correctness of data storage verification in the cloud becomes even more challenging.

Again, cloud computing is not only a third party data warehouse. Whereas the data stored in the cloud may be frequently updated by the users, in the form of insertion, deletion, modification, appending, reordering, etc. To ensure the security correct storage it is necessary to check data stored is correctly.

As security of data storage is important then security of data forwarding in the cloud is also important. To achieve security for data transfer in cloud, we introduce notion of conditional proxy re-encryption whereby only the cipher text satisfying one or more condition set by user one can be transferred by proxy and decrypted by user two.

2. Literature Survey

In this section we briefly review the cloud storage system, proxy re-encryption schemes. In [1],[15], Hsia-Ying Lin and Wen-Guey Tzeng et al gives an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. They use erasure correcting code in the file distribution preparation to provide redundancies and guarantee the data dependability and this construction drastically reduces the communication and storage overhead as compared to the traditional replication-based file distribution techniques. By using identifiers with verified distributed erasure-coded data, their scheme achieves the storage correctness insurance as well as data error localization: whenever corrupted data has been occurred during the storage correctness verification, their scheme can almost guarantee the identification of the misbehaving server(s), [11], [14].

In [1], they propose a threshold proxy re-encryption scheme and integrate it with a decentralized erasure code such that a secure distributed storage system formed. The distributed storage system provides secure data storage and data retrieval, as well as enables user to forward his data in the storage servers to another user without retrieving the data back. They use the proxy re-encryption scheme because this was helpful for encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. Their method was fully integration of encoding, encryption, and forwarding which makes the storage system efficient and fulfill the requirements of data robustness, data confidentiality, and data forwarding.

In [2],[7] Hsia-Ying Lin and Wen-Guey Tzeng et al. firstly encodes the message, then message encrypted to store in cloud storage system. They used decentralized reassure code for the purpose of data storage in the cloud storage system.

2.1 Proxy Re-encryption

Initially the concept proxy re-encryption was proposed by Mambo and Okamoto et al., [3]. In a proxy re-encryption scheme, the proxy cryptosystem allows an original decryptor to transform its ciphertext to a ciphertext for a designated decryptor, proxy decryptor. As the ciphertext transformation is over, the proxy decryptor can find a plaintext in place of the original decryptor. Such a cryptosystem is very useful for large amount of decrypting operation. This type of cryptosystem can actually speed-up the decrypting operation by authorizing multiple proxy decryptors. Blaze, Bleumer, and Strauss (BBS) et al.[4],[8],[17], proposed the concept of proxy re-encryption in which they provides a semi trusted proxy which converts a ciphertext for user one into a ciphertext for user two without seeing the plaintext and it becomes bidirectional conversion,[12],[13].

In [5], author Tang et al. proposed Type-based proxy re-encryption scheme which gives access rights of a re-encryption keys to users. According to these rights, a user can decide what type of message and with whom he wants to share this proxy re-encryption scheme. In [6],[16], author Ateniese et al. provide a proxy re-encryption scheme in such a way that for a given re-encryption keys , a proxy server cannot find out the identity of the recipient. Ateniese et al. proposed unidirectional proxy re-encryption scheme.

3. Proposed Work

For more securely data transfer in cloud we introduce conditional proxy re-encryption. We form the security model of conditional proxy re-encryption. An efficient construction of conditional proxy re-encryption scheme offers several advantages over previous systems including chosen-cipher text security, uni directionality and collusion-resistance [10]. This scheme has better overall efficiency in terms of both computation and communication cost and provides better security.

In cloud, instead of converting all cipher texts, user may only want the proxy to convert the cipher texts with a specific word, such as ''Important''. This problem solved by introducing the notion of conditional proxy re-encryption in construction of secure cloud storage. The conditional proxy re-encryption means user1 sends some message to user 2 and he wants that the user2 access some important messages earlier as these messages are very important. So user1 for important messages set condition i.e. set w= "important" with whom to send and is the reply is needed for this message, in subject and encrypt message. Condition is set in the subject because user 1 wants only the subject of message is visible to the proxy and not the body of the message, so message is encrypted by user 1. This message re-encrypted by proxy server using re-encryption key which formed by user1 and user 2 secret key and conditional key provided by user1 and converts into the ciphertext. User1 wants only user2 read this ciphertext that satisfying condition w="important" rather than all other user 1's ciphertext. Also user1 and user2 do not want that proxy server able to read this condition set ciphertext for security purpose. If User 2 is not receiving or replying for this message then proxy check for another condition which is set by user 1.

Here user 1 provides set of conditions to the proxy for difficult circumstances whenever occurs. So proxy having set of conditions, on that basis proxy decides if user 2 is not replying then he checks for the next condition from the set. This condition may be tell proxy to send this message to another user i.e. user 3 on behalf of user 2.So here proxy server have functionality of partial decryption, means in difficult circumstances proxy are able to or having right to decrypt that message on behalf of user which is actually receiver of that message. At that time proxy decrypts message and re- encrypt that message with key of another user called user 3 and send that message to user 3.Proxy server are connected with storage server and key server for accessing multiple conditions which are provided by owner of the message which are actually stored on storage server and keys which are managed by key server. So proxy server having all the information needed for encryption, condition checking and data forwarding to whom and which data to be forward and what to do when difficult situation occurs. If condition is not set properly by user1, then proxy is not able to re-encrypt the message and not able to forward this message towards another user. At that time, proxy replies to the user 1 with error message. So under right condition proxy re-encrypts the message efficiently and able to forward that message. This will be the advantage of this scheme to achieve more security.

4. Definitions

In this section we briefly review some algebraic settings, assumptions which are considered in the creation of structure of conditional proxy re-encryption for constructing secure cloud storage.

4.1 Bilinear maps

Let G1 and G2 be multiplicative cyclic groups of prime order p, and g be a generator of G_1 . We say

e : $G_1 \times G_1 \rightarrow G_2$ is a bilinear map [9] , if the following conditions hold.

(1) $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ For all a, b Є Zp and $g_{1,} g_2$ Є G_1

(2) e(g, g) ≠ 1

(3) There is an efficient algorithm to compute e($g_1 g_2$)for all g_1, g_2 Є G_1

4.2 Computational bilinear Diffie-Hellmen assumption

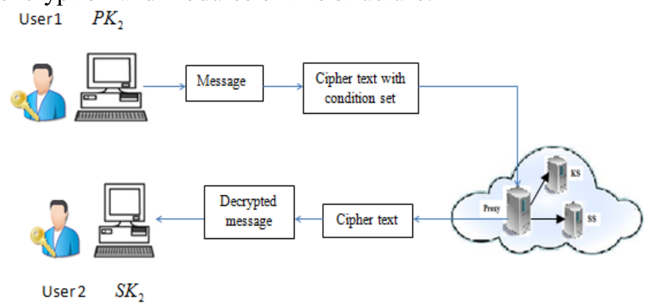
Let G1 and G2 be multiplicative cyclic groups of prime order p. Let e: $G_1 \times G_1 \rightarrow G_2$ be a bilinear map and g be a generator of G_1 . The CBDH problem in (G_1, G_2, e) is as follows:

Given (g , $g^{\frac{a}{b}}, g^{a}, g^{b}, g^{c}$) for a, b, c Є Z_P^* , compute w= e $(g, g)^{abc}$ ϵ G_2 .An algorithm A has an advantage Є in solving CBDH in (G_1, G_2, e) if

Pr [A (g , $g^{\frac{a}{b}}, g^{a}, g^{b}, g^{c}$)= e $(g, g)^{abc}$]>= Є
Where the probability is over the random choice of a, b, c Є Z_P^*, the random choice of g Є G_1 and the random bits of A.

5. Structure and modules

In this section we briefly review the structure of the cloud storage with data forwarding using conditional proxy re-encryption and modules of this structure.



SS = Storage Server   KS = Key Server,   Proxy = Proxy Server,

PK_2= Public key of user 2, SK_2= secret key of user 2

Figure 1: Structure of Cloud Storage with Safe Data Forwarding

In the presented structure actual encryption done by owner of the message with some conditions set on the subject of the message which is to be send. The message converted into cipher text with condition set passes through the cloud (proxy server) to the receiving user. The proxy server responsible for re-encryption of message, condition checking, data forwarding in cloud and managing information about set of conditions and handling the difficult situations like receiver of the message is on vacation or not replying for the given message, management of key server and storage servers. Here proxy server does many of the functioning on behalf of user so user has not to worry about data transfer and storage done. The conditional encryption is used for giving rights to users as proxy server decides which message will be send to whom on the basis of multiple conditions which are set on the message. According to that proxy sends message to the expected users of that message.

This structure is performed in the following modules:

5.1 Key Generation: In this for each user pair of secret key and public key are generated which is required for encryption and decryption of the given important message. For re-encryption by proxy server, re-encryption key is created using secret keys of both users who want to send message and who want to receive message. Condition key is created by owner of the message at the time when condition is set to the message and provides condition key to the proxy server with set of conditions which are useful in checking another condition when one condition fails due to some reason to forward the message without any error.

5.2 Encryption: Encryption of message done by owner of the message by taking key of the receiving user with some condition set in subject field of the message. The Encryption is performed on the basis of standard encryption algorithm EIGammal which provides good encryption facility.

5.3 Re-Encryption: Re-encryption done by the proxy server on the encrypted message with set condition w by owner of the message i.e. user 1.Re-encryption is performed using the algorithm Diffie–Hellman key exchange. By using this algorithm a share key is formed using senders and receivers secret key on which both are agreed. This key is provided to proxy server for re-encrypting the message. Here we use share key with conditional key for re-encryption for achieving better encryption facility. Testing of conditions which are set by owner of message are done at the time of re-encryption by proxy server using the set of conditions provided by the owner. As any condition is not satisfied or not matched to the set of conditions, then proxy server not able to forward that message to the receiver. At that time proxy server sends error message to the owner of the message and tell to set right and accurate condition on the message.

5.4 Decryption: In this original message obtained to the expected receiver. Receiver uses only his/her secret key to decrypt the message. Here the receiver does not require any conditional key to decrypt the message as he/she does not having any right to access the conditional key.

6. Algebraic Calculations

In this section we briefly review the actual mathematical design of the given structure.

6.1 Creation of generator

Suppose p is prime, Find some α Є Z_P such that each number $\beta \epsilon Z_P^* (= Z_P \setminus \{0\})$ can be written as $\beta - \alpha^a (mod\ p)$ for some a, where is called primitive element of Z_P .

If P = 17 then α = 6 is a primitive element of Z_{17} .This is because Z_{17} we have

Z_{17} = {1= 6^{16} mod 17, 2= 6^2 mod 17, 3= 6^{15} mod 17…..}

6.2 Encryption

Let the public group G and an element α Є G of order N
Let G1 = < α > where α= generator
Let key space = G1
For each key k Є K, the plaintext and cipher text space are

$M_k = G$

$C_k = G_1 \times G = \{(\beta_1, \beta_2) : \beta_1 \in G_1, \beta_2 \in G\}$

The randomized set is $R_k = Z_N$

For each key k Є K, the encryption function

$E_K : M_k \times R_k \to C_k$ is given by

$$E_K(m,r) = (a^r, k^r.m)$$

For each key k Є K, Private Key = integer d Є Z_N such that $k = \alpha^d$ and decryption function $D_k : C_k \rightarrow M_k$ is given by

$$D_k(C_1, C_2) = C_2.(C_1^d)^{-1}$$

6.3 Re-encryption

Given public group G and an element α Є G of order N
Given Z_N ={a, b, c,………,n-1}
Steps:
1. Let choose any random integer a, where a Є Z_N then compute $K_A = \alpha^a$
2. Let choose any random integer b, where b Є Z_N then compute $K_B = \alpha^b$
3. Then calculate share key K from given keys, where a, b Є Z_N, Calculate K= α^{ab} or α^{ba} .

6.4 Conditional Encryption

Let choose random integer a, where a Є Z_N
Given Z_N = {a, b, c,………,n-1}
Let C= G {m, a, w, C_k }
Where m= message to be encrypted,
a = random integer chosen as a secret key of user,
w= condition given to the text file, here on condition we decide rights given to access the message,
C_k = cipher text with condition set
Such that p= {if w= w then $C_k = m^a.w$ and
if w ≠ w then $C_k = \perp$}
Where ⊥= error message

7. Security Model

In this section we check the security of message and privacy of condition. Here, the opponent is allowed to get the plaintexts of almost all cipher texts except for a specified cipher text. Then security notion guarantees that the opponent can take any trapdoors, except the ones that are associated with the specific condition, and further, it should not be able to decide which condition corresponds to the provided cipher text. This security notion guarantees that only the one who has the private key can decrypt cipher texts.

8. Conclusion

As security of data storage is important, also the security of data transfer is important. We achieve the security of data transfer by introducing the conditional proxy re-encryption. It provides many advantages like chosen cipher text attack, uni directionality and collusion-resistance over the previous schemes. This scheme provides secure model of cloud storage with safe data forwarding.

References

[1] Hsiao-Ying Lin, Wen-Guey Tzeng, "A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding", IEEE Computer Society,vol.23, No.6, June 2012.

[2] H.-Y. Lin and W.-G. Tzeng, "A Secure Decentralized Erasure Code for Distributed Network Storage," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 11,pp. 1586-1594, Nov. 2010.

[3] M. Mambo and E. Okamoto, "Proxy Cryptosystems: Delegation of the Power to Decrypt Ciphertexts," IEICETrans. Fundamentals of Electronics, Comm. and Computer Sciences, vol. E80-A, no. 1, pp. 54- 63, 1997.

[4] M. Blaze, G. Bleumer, and M. Strauss, "Divertible Protocols and Atomic Proxy Cryptography," Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT), pp. 127-144, 1998.

[5] Q. Tang, "Type-Based Proxy Re-Encryption and Its Construction," Proc. Ninth Int'l Conf. Cryptology in India: Progress in Cryptology (INDOCRYPT), pp. 130-144, 2008.

[6] G. Ateniese, K. Benson, and S. Hohenberger, "Key-Private Proxy Re-Encryption," Proc. Topics in Cryptology (CT-RSA), pp. 279-294, 2009.

[7] A.G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized Erasure Codes for Distributed Networked Storage," IEEE Trans. Information Theory, vol. 52, no. 6 pp. 2809-2816, June 2006.

[8] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security, vol. 9, no. 1, pp. 1-30, 2006.

[9] D. Boneh, X. Boyen, Efficient selective-ID based encryption without random oracles, in: Proc. of EUROCRYPT 2004, in: LNCS, vol. 3027, Springer, Heidelberg, 2004, pp. 223–238.

[10] R. Canetti, S. Hohenberger, Chosen-ciphertext secure proxy re-encryption, in: Proc. of the 14th ACM Conference on Computer and Communications Security, ACM, New York, NY, USA, 2007, pp. 185–194.

[11] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,"Proc. IEEE 29th Int'l Conf. Computer Comm. (INFOCOM), pp. 525-533, 2010.

[12] S. Chow, J. Weng, Y. Yang, R. Deng, Efficient unidirectional proxy re-encryption, in: Proc. of AFRICACRYPT 2010, in: LNCS, vol. 6055, Springer,Heidelberg, 2010, pp. 316–332.

[13] A. Shamir, "How to Share a Secret," ACM Comm., vol. 22, pp. 612-613, 1979.

[14] G. Ateniese, S. Kamara, and J. Katz, "Proofs of Storage from Homomorphic Identification Protocols," Proc. 15th Int'l Conf.Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 319-333, 09.Communications Security, ASIACCS 2009, 2009, pp. 322–332.

[15] A.G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Ubiquitous Access to Distributed Data in Large-Scale Sensor Networks through Decentralized Erasure Codes,"

Proc. Fourth Int'l Symp. Information Processing in Sensor Networks (IPSN), pp. 111- 117, 2005.

[16] T. ElGamal. A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In Advances in Cryptology-Crypto'84, LNCS 196, pp.10-18, Springer-Verlag, 1984.

[17] Y. Dodis, and A.-A. Ivan. Proxy Cryptography Revisited.In Proc. of NDSS'03, 2003.

Author Profile

Tushar Rane received the B.E degree in Computer Engineering from SRES College of Engineering, Kopargaon in 2000 and M.E degree in Computer Engineering from Bharati Vidyapeeth University, Pune in 2009. He is pursuing PhD in Computer Engineering from Vishwakarma Institute of Technology, Pune. During 2001-2002 and during 2002-2005 he worked as a Lecturer in Amrutvahini College of Engineering, Sangamner and in SRES College of Engineering, Kopargaon. He became Assistant Professor in Pune Institute of Computer Technology from 2005 to present. Presently he is working in cloud Live Migration of Virtual Machines.

Shrishail Patil completed his education from Jain Gurukul Solapur.He received PhD degree in Computer Engineering. He worked as Lecturer in Solapur Education Society**,** Pune during 1989-2003, Assistant Professor at Sharada Group of Institutions, Agra Area for 6 months, Professor in Computer Engineering at Bharati Vidyapeeth, Pune from 2004 to 2009, and Principal at JSPM Imperial college of Engineering and Research, Pune during 2009-2011.Currently he became Professor in Computer Engineering at Vishwakarma Institute of Technology, Pune.

Anita Khade received B.Tech degree in Computer Engineering from Dr. Babasaheb Ambedkar Technological University, Lonere in 2009.Pursuing M.E in Information Technology from Pune Institute of Computer Technology, Pune. Presently she is working on cloud security.