ISSN No : 2454-423X (Online)

International Journal of Research in Advanced Computer Science Engineering

A Peer Reviewed Open Access International Journal

www.ijracse.com

Improved Schemes to Secure Distributed Data Storage against Untrusted Users

K.Nagaraju

Department of Computer Science & Engineering, Aditya Institute of Technology and Management, Srikakulam District, Andhra Pradesh 532201, India.

R.Jayamma

Department of Computer Science & Engineering, Aditya Institute of Technology and Management, Srikakulam District, Andhra Pradesh 532201, India.

ABSTRACT:

"Proxy servers" are the important thing we are using here to secure the data from untrusted users by assuming them(proxy servers) as trusted storage. Secure distributed data storage can shift the burden of maintaining a large number of files from the owner to proxy servers. Proxy servers can convert encrypted files for the owner to encrypted files for the receiver without the necessity of knowing the content of the original files. The original files will be removed by the owner for the sake of space efficiency. Hence, the issues on confidentiality and integrity of the outsourced data must be addressed carefully. we propose two identity-based secure distributed data storage schemes. Our schemes can capture the following properties. The file owner can decide the access permission independently without the help of the private key generator .For one query, a receiver can only access one file, instead of all files of the owner. Our schemes are secure against the collusion attacks, and from untrusted users namely even if the receiver can compromise the proxy servers, he cannot obtain the owner's secret key.Although these schemes are secure against different types of attacks and all the access permissions made by the owner himself.

Key words: ***Distributed Data Storage, Identity-based System, Access Control, Security, integrity and confidentiality.***

1. INTRODUCTION:

Cloud computing provides users with a convenient mechanism to manage their personal files with the notion called database-as- a-service (DAS). In DAS schemes [1-5], a user can outsource his encrypted files to untrusted proxy servers. Proxy servers can perform some functions on the outsourced ciphertexts without knowing anything about the original files. Unfortunately, this technique has not been employed extensively. The main reason lies in that users are especially concerned on the confidentiality, integrity and query of the outsourced files as cloud computing is a lot more complicated than the local data storage systems, as the cloud is managed by an untrusted third party. After outsourcing the files to proxy servers, the user will remove them from his local machine. Therefore, how to guarantee the outsourced files are not accessed by the unauthorized users and not modified by proxy servers is an important problem that has been considered in the data storage research community. Furthermore, how to guarantee that an authorized user can query the outsourced files from proxy servers is another concern as the proxy server only maintains the outsourced ciphertexts. Confidentiality is proposed to prevent unauthorized users from accessing the sensitive data as it is subject to unauthorized disclose and access after being outsourced.

Since the introduction of DAS, the confidentiality of outsourced data has been the primary focus among the research community. To provide confidentiality to the outsourced data, encryption schemes are deployed

Cite this article as: K.Nagaraju & R.Jayamma, "Improved Schemes to Secure Distributed Data Storage against Untrusted Users", International Journal of Research in Advanced Computer Science Engineering, Volume 5 Issue 3, 2019, Page 1-6.

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Integrity can prevent outsourced data from being replaced and modified. Some schemes have been proposed to protect the integrity of the outsourced data, such as proof of retrievability and provable data possession . These schemes, digital signature schemes and message authentication codes (MAC) are deployed. Query in data storage is executed between a receiver and a proxy server. The proxy server can perform some functions on the outsourced cipher texts and convert them to those for the receiver. As a result, the receiver can obtain the data outsourced by the owner without the Proxy server knowing the content of the data.

2. DATA STORAGE SYSTEM:

From fig,Data storage systems enable users to store their data to external proxy servers to enhance the access and availability, and reduce the maintenance cost. The privacy issues in data outsourcing expanding from the data confidentiality to data utility, and pointed out the main research directions in the protection of the externally stored data. According to that, the data storage systems comprehensively and classified them into three kinds based on their security services: networked file systems (NFS), storage-based intrusion detection systems (SBIDS) and cryptographic file systems (CFS) [7].

Networked File Systems:

In these systems, proxy servers are assumed to be trusted. They authenticate receivers and validate access permissions. The interactions between the proxy servers and receivers are executed in a secure channel. These systems cannot provide an end-to-end data security, namely they cannot ensure the confidentiality of the data stored at the proxy server according to the schemes, a receiver authenticates himself to the proxy server using his password. The proxy server passes the authentication result to the file owner. The owner will make an access permission according to the received information.

Storage-based Intrusion Detection Systems:

In these systems, an intrusion detection scheme is embedded in proxy servers or the file owner to detect the intruder's behaviors, such as adding backdoors, inserting Trojan horses and tampering with audit logs. These schemes can be classified into two types: host-based system and network-based system. In the host-based systems, an intrusion detection scheme is embedded in the host to detect the local intrusion actions. On the contrary, in network-based systems, an intrusion detection scheme is embedded in the proxy servers to detect the external intruder's actions. The main advantage of these systems is that proxy servers can still detect the intrusion actions even if the host is compromised as the proxy server are independent from the host [9] .

Cryptographic File System:

In these systems, an end-to-end security is provided by cryptographic protocols which are executed by the file owner to prevent proxy servers and unauthorized users from modifying and accessing the sensitive files. These systems can be divided into two types: shared file system and non-shared system. In shared file systems, the owner can share his files with a group of users. Cryptographic techniques deployed in these systems are key sharing, key agreement and key revocation. In non-shared file systems, in order to share a file with another user, the owner can compute anaccesskey for the user using his secret key. In these two systems, the integrity of the sensitive files is provided by digital signature schemes and message authentication codes (MAC).

3. IDENTITY BASED PROXY RE_ENCRYPTION:

An atomic proxy cryptosystem where a semi-trusted proxy server can transfer a ciphertext for the original decryptor to a ciphertext for the designated decryptor without knowing the plaintext. Proxy cryptosystem as an efficient primitive has been used in email forwarding, law enforcement and data storage. Identity based cryptosystem is a system, where the public key can be any arbitrary string and the secret key is issued by a trusted party called the private key generator (PKG). Being different from public key infrastructure (PKI), two

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

parties can communicate directly without verifying their public key certificates in identity-based systems. In the IBE, IBPE schemes the master secret key which is used to extract secret keys for users is split into two parts.

One is sent to the proxy server and the other is sent to the user. The user can decrypt a ciphertext for him with the help of the proxy server. Pointed out that these schemes are not secure against the collusion attacks, namely the master secret key can be exposed if the user can compromise the proxy server. The first identity-based proxy re-encryption where the proxy sever can transfer a ciphertext for the original decryptor to a ciphertext for the designated decryptor after he gets a re-encryption key from the former. We divide the IBPRE schemes into the following two types based on the generation of the re-encryption key:

The re-encryption key can be computed by the original decryptor .In these schemes, for a decryption request, the original decryptor selects a random number and computes a re-encryption key by randomizing his secret key. Then we encrypts the selected random number under the receiver's identity. Finally, sending the re-encryption key and the ciphertext to the proxy server. Using the re -encryption key, the proxy server can transfer a ciphertext for the original decryptor to a ciphertext for the designated decryptor. The designated decryptor decrypts the ciphertext using his secret key and obtains the random number selected by the original descriptor. So we can decrypt the re-encrypted ciphertext by the random number. Unfortunately, these schemes are vulnerable to the collusion attacks. If the designated decryptor can compromise the proxy server, they can decrypt the ciphertext, obtain the random number selected by the original decryptor and compute the secret key of the original decryptor. The re-encryption key must be computed by the PKG In these schemes, the PKG computes the re-encryption key by checking the secret keys of the original decryptor and the designated decryptor [11].

4. INDENTITY- BASED SECURE DISTRIBUTED DATA STORAGE:

In an identity-based secure distributed data storage (IBSDDS) scheme, a user's identity can be an arbitrary string and two parties can communicate with each other without checking the public key certificates. At first, the file owner encrypts his files under his identity prior to outsourcing them to proxy servers. Later owner sends the ciphertexts to the proxy servers. The proxy servers can transfer a ciphertext encrypted under the identity of the owner to a ciphertext encrypted under the identity of the receiver after they has obtained an access permission (re-encryption key) from the owner. To provide confidentiality for the outsource data, an efficient IBSDDS scheme should provide the following properties [10].

Unidirectional:
After receiving an access permission, the proxy server can transfer a ciphertext for Alice to a ciphertext for Bob while he cannot transfer a ciphertext for Bob to a ciphertext for Alice.

Non-interactive:
The access permission can be created by the file owner without any trusted third party and interaction with him.

Key optimal:
The size of the secret key of the receiver is constant and independent of the delegations which he accepts.

Collusion-safe:
The secret key of the file owner is secure even if the receiver can compromise the proxy server.

Non-transitive:
Receiving the access permissions computed by Alice for Bob and Bob for Charlie, the proxy server cannot transfer a ciphertext for Alice to a ciphertext for Charlie.

File-based access:
For one query, the receiver can only access one file. This can improve the security of the outsourced files and is desirable to maintain the access record. Proxy invisibility discussed in n on-transitive is difficult to achieve as the length of the re-encrypted ciphertext is subject to be different from that of the original

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

ciphertext. Namely the secret key is created by the PKG, instead of the user. Hence, the file owner in an IBSDSS scheme has less control on his secret key than that in other public key encryption schemes. Although IBPRE holds partial properties of IBSDDS, it cannot be used in IBSDDS systems directly. For example, in the current IBPRE schemes, the receiver and the proxy servers can cooperate to access all the files outsourced by the owner as the access permission (re-encryption key) is only bound to the identity of the receiver and independent of the file. This is undesirable for the file owner to record the accessed number of his files. Since the PKG can generate a secret key for each user, he can decrypt the ciphertexts and obtain the original files if he knows the identity used to encrypt the files. Therefore, in this paper, we assume that the PKG [8] is honest and can be trusted by all users in the systems.

5. ANALYSING DISTRIBUTED DATA STORAGE SCHEMES:

The distributed data storage schemes analysis task is going to establish complete information about the concept, behavior and the other constraints like performance measure and the system optimization. The main goal of distributed data storage schemes Analysis is to completely specify the technical details for the main concept in a concise and unambiguous manner [6].

Architecture of proposed system:

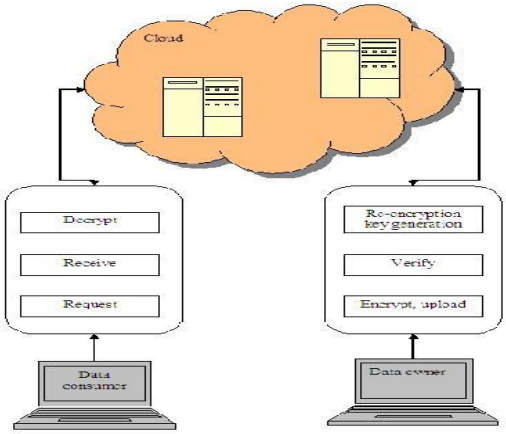

Fig: Distributed storage system in cloud environment.

Architectural Diagram Analysis is the process of understanding the environment in which a proposed system or systems will operate and determining the requirements for the system and designed in 4-modules.

Advantages this Proposed system:

In proposed system the user can upload the data and download the required data. Here both sender and receiver can upload and access the data by registering their names with their ID's. These ID's are going to store in a proxy servers and invisible.

1. The user can act as both data owner and data consumer.
2. Any user can get the required data by registering with them.
3. The proposed system is going to provide more confidentiality.

Network Deployment module:

This module contains details about cloud based network. The cloud contains proxy servers and private key generator. The proxy server is connected to public key generator. Then the users are connected to cloud in wireless link. Then the every user makes registration to proxy server. Then the proxy server generates secret key using private key generator and distribute to user. Then the user stores the key. Then every node sends their signature to proxy server. In this the user can act as a data owner and data consumer [4].

Uploading the data module:

When we are uploading the data, every node can act as data owner and data consumer. Here the data up loader is called as data owner and the data user is called as data consumer. The data owner first takes parameter, identity, and message and encrypt the message and sent to proxy server. Then the proxy server validates the cipher text and store in database.

Process of file requests module:

Whenever the data consumer needs one particular file then the user (data consumer) sent request by entering

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

the file name in text field. While entering the file name, that particular filename only should visible. Then sent request with Authentication information to proxy server. Then the proxy server verifies the signature and re-directs the data to data owner.

Response of data owner module:
After receiving the request, the data owner checks the Authentication information and takes public parameter, receiver's identity, owner's secret key as input and generates re-encryption key and sent to proxy server. The proxy server takes receiver's id, cipher text and re-encryption key, and outputs a cipher text. Then the proxy server sends the cipher text to user. Then the receiver takes the public parameters, receiver's secret key, and re-encrypted cipher text and decrypts the cipher text and view.

Algorithms and techniques used:
We need so many algorithms the basic and important algorithms of all algorithms we are using here is:

Setup Algorithm: The setup algorithm takes as input a security parameter 1_, and outputs the public parameters and a master secret MSK.

KeyGen: The key generation algorithm takes as input the public parameters, an identity ID and the Master Secret Key MSK, and outputs a secret key S for the identity ID.

Encryption: This algorithm Suppose that there are k messages {M1,M2, ・・・,Mk}. To encrypt the message Mi, the encryption algorithm takes as input the public parameters, the identity ID and the message Mi, and outputs the cipher text= 1, 2, ・・・, k. It sends the cipher texts to the proxy servers.

Query algorithm: It takes as input the receiver's identity ID_, the receiver's secrete key and the cipher text, and outputs an authentication information AI. It sends (ID_,AI,CT) to the proxy server. The proxy server redirects (ID_,AI,Ci,2) to the owner with identity ID.

Permission algorithm: This Algorithm checks the authentication information AI. If the receiver is legal, this algorithm takes as inputs the public parameters, the receiver's identity ID_ and the owner's secret key, and outputs an access permission (re-encryption key) .

Re-encryption: The re-encryption algorithm takes as input the public parameters, the receiver's identity ID_, the access permission and the cipher text, and output cipher text.

Decryption: There are two algorithms. One is for the owner and the other is for the receiver.

Decryption
The owner decryption algorithm takes as input the public parameters, the owner's secret key and the cipher text, and outputs the message Mi.

Decryption
The receiver decryption algorithm takes as input the public parameters, the receiver's secret key and the re-encrypted cipher text.

Proxy Re-encryption Algorithm: This algorithm is going to take the input as the public parameters, the receivers ID, and Cipher text that may be the owners (or) receivers and gives the output as Cipher text to the receivers with ID's.

RSA: This algorithm generates two keys. One is public key and another is private key.In that, public key is used to encrypt the data and private key is used to decrypt the encrypted file.

Signature: Signature is nothing put just a verification key. So to generate key, we are key generation method. This method is available in RSA algorithm. This method generates one key for every user.

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

CONCLUSION:

Distributed data storage schemes provide the users with convenience to outsource their files to untrusted proxy servers. Identity-based secure distributed data storage(IBSDDS) schemes are a special kind of distributed data storage schemes where users are identified by their identities and can communicate without the need of verifying the public key certificates. In this paper, I proposed two new IBSDDS schemes in standard model where, for one query, the receiver can only access one file, instead of all files. Furthermore, the access permission can be made by the owner, instead of the trusted party. Our schemes are secure against the collusion attacks. The first scheme is CPA secure, while the second one is CCA those are made by untrusted users.

REFERENCES:

[1].A. Juels and B. S. K. Jr., "PORs: Proofs of retrievability for large files," in Proceedings: ACM Conference on Computer and Communications Security - CCS'07 (P. Ning, S. D. C. di Vimercati, and P. F. Syverson, eds.), (Alexandria, Virginia, USA), pp. 584–597, ACM, Oct. 2007.

[2]."Compact Proofs of Retrievability" by Hovav Shacham and Brent Waters.

[3]. C.-K. Chu and W.-G. Tzeng, "Identity-based proxy re-encryption without random oracles," in Proc. Information Security Conference - ISC'07 (J. A.Garay, A. K. Lenstra, M. Mambo, and R. Peralta, eds.), vol. 4779 of Lecture Notes in Computer Science, (Valparaso, Chile), pp. 189–202, Springer, Oct. 2007

[4]."Executing SQL over Encrypted Data in the Database Service Provider Model" by Hakan Hacıg umus¸ Bala Iyer, Chen Li_ Sharad Mehrotra.

[5]. L. Bouganim and P. Pucheral, "Chip -secured data access: Confidential data on untrusted servers," in Proc. International Conference on Very Large Data Bases - VLDB'02, pp. 131–142, Morgan Kaufmann, Aug. 2002.

[6].B. Carbunar and R. Sion, "Toward private joins on outsourced data," IEEE Transactions on Knowlege and Data Engineering, vol. 9, no. 24, pp. 1699–1710, 2012.

[7] J. Hur, "Improving security and efficiency in attribute-based data sharing," IEEE Transactions knowlege and Data Engineering,p. Digital Object Indentifier 10.1109/TKDE.2011.78.

[8] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 7, pp. 1214–1221 ,2011.

[9]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proc. ACM Conference on Computer and Communications Security-CCS'07 (P. Ning, S. D. C. di Vimercati, and P. F. Syverson, eds.), (Alexandria, Virginia, USA), pp. 598–610, ACM, Oct. 2007.

[10] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. International conference on Security and privacy in communication netowrks - SecureComm'08, (Istanbul, Turkey), Sep., ACM, 2008.

[11] C. C. Erway, A. K¨upc¨u, C. Papamanthou, and R. Tamassia "Dynamic provable data possession," in Proc. ACM Conference on Computer and Communications Security - CCS'09 (E. Al-Shaer, S. Jha, and A. D. Keromytis, eds.), (Chicago, Illinois, USA), pp. 213–222, ACM, Nov. 2009.