International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 03 | Mar-2016 www.irjet.net p-ISSN: 2395-0072

Secure Sharing of Data in the Cloud Environment

[1]R. Anitha, [2]V. Suganya

[1]Assistant Professor, Department of Computer Science and Engineering
[2]P.G. Scholar,Department of Computer Science and Engineering,
Valliammai Engineering College, Tamil Nadu, India.

Abstract -*Cloud storage is an application of clouds that delivers organizations from establishing in house information storage systems. However, cloud storage provides rise to security considerations. In case of group-shared data, the information face each cloud-specific and conventional insider threats. Secure information sharing among a group that counters insider threats of legitimacy, however malicious users are an important analytical issue. During this paper, we propose the Secure sharing of data in the Clouds methodology that provides: 1) information confidentiality and integrity, 2) access control, 3) information sharing (forwarding) without misusing compute-intensive re-encryption, 4) forward and backward access control, 5) insider threat security and 6) key management. The secure sharing of data methodology encrypts a file with one encryption key. Two totally different key shares for every of the users are generated, with the user only obtaining one share. The possession of a single share of a key permits the secure sharing of data methodology to counter the insider threats. The other key share is kept by a trusted third party, which is termed as a crypto graphical server. The Sharing of data in the clouds methodology is applicable to standard and mobile cloud computing environments. We have a tendency to implement an operating prototype of the secure sharing of data in the cloud methodology and measure its performance supported the time consumed throughout numerous operations.*

Key Words: Cloud computing, Key management, cryptographic keys, Trusted Third party, Cloud Storage.

1. INTRODUCTION

Cloud computing is quickly rising because of the provisioning of elastic, flexible, and on-demand storage and computing services for customers. [1] Organizations with a low budget will currently utilize high computing and storage services without heavily investing in infrastructure and maintenance. However, the loss of management over data and computation raises several security issues for organizations, thwarting the wide ability of the general publiccloud [2].

In this method, we present the method of proposed methodology secure sharing of data in the cloud that secure data sharing and data forwarding among a group without involving re-encryption in the cloud environment. The three methods are used in this model. They are users, cloud and Cryptographic Server. Users: The users are the clients of the storage cloud. For each file, one user is going to be the owner of the file, whereas the others within the cluster are going to be the info consumers. The owner of the file decides the access rights of the opposite cluster members. The access rights are granted and revoked supported the decision of the owner. The access rights are managed by the cryptographic servers within the form of an access control list file. A separate Access control list is maintained for every of the data files. Clouds: The cloud provides storage services to the user. The data on the cloud have to be compelled to be secured against privacy breaches. The confidentiality of the information is ensured by storing encrypted data over the cloud. The cloud within the secure sharing of data methodology only involves basic cloud operations of file upload and transfer. Therefore, no changes at the protocol or implementation level of the cloud are needed.

Cryptographic Server: The Cryptographic server could be a sure party and is liable for security operations, such as key management, encryption, decryption, the management of the Access control list for providing confidentiality, and security data forwarding among the cluster. The users of Secure sharing of data required to be registered with the Cryptographic server to get the safety services. The cryptographic server is assumed to be a secure entity within the projected methodology. The cryptographic server may be maintained by an organization or can be owned by a third-party supplier. However, the Cryptographic server maintained by an organization can generate a lot of trust within the system.

1.1 Related work

To projected a certificate less proxy re-encryption (CL-PRE) theme for securely sharing the information among a group in the public cloud. Within the CL-PRE theme, the info owner encrypts the information with the symmetric key, Xueat al [3]. Afterwards, the symmetric key is encrypted with the general public key of the information owner. Each the encrypted information and also the key upload to the cloud. The encrypted key is re-encrypted by the cloud (that acts as a proxy re-encryption agent) that becomes decryptable by the user's non-public key. The public–private keys generated within the proposed theme are not supported the certificates. The user's identity is employed to generate the public–private key combines.

The proxy re-encryption is based on linear pairing and also the BDH that makes the CL-PRE scheme computationally intensive. The computed value of the linear pairing is high as compared with the quality operations in finite fields. To reduce the processing overhead of bilinear pairing, SEO et al. [4] Introduced a mediate certificateless encryption approach for information sharing within the public cloud that avoids bilinear pairing. Within the proposed theme, the cloud generates the public–private key pairs for all of the users and transmits the public keys to all or any of the participating users. Partial secret writing is performed in the cloud. Due to the actual fact that key management and partial decryption are handled by the cloud, user revocationis easier to handle. However, the proposed theme treats the public cloud each as a trusty and untrusted entity at an equivalent time. From a security perspective, it is not recommended to shift the key generation method to the shared multitenant public cloud atmosphere. Moreover, the decryption is performed twice within the system that reduces the advantage of not pairing to some extent..

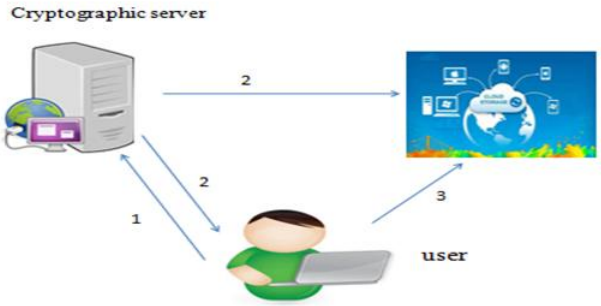
Khan et al. [5] additionally used the El-Gamal crypto system and additive pairing for the sharing of sensitive data in the cloud. Moreover, the proposed scheme in [5] used the concept of progressive cryptography that divides the info into blocks and incrementally encrypts the blocks. The proposed scheme uses a trusted third party as a proxy that performs the compute-intensive operations of key generation, re-encryption, and managing access to the info. However, the procedural complexities of bilinear pairing still exist within the system. Chen and Tzeng [6] proposed a strategy supported the shared key derivation methodology for securing information sharing among a group. The methodology uses a binary tree for the computation of the keys. However, the process price of the proposed scheme is high because the re-keying mechanism is heavily utilized in the planned scheme. Moreover, the theme is not tailored for public cloud systems as a result of certain operations need centralized mediations. An analogous Rivest–Shamir–Adleman (RSA)-based approach was additionally proposed in [7]. However, the scheme was vulnerable against collusion attacks. The secure sharing of data methodology, which is proposed during this paper, securely shares the information among a group while not using the El-Gamal cryptosystem, the BDH, and linear pairing. The secure sharing of data methodology relies on symmetric cryptography without re-encryption. The same properties avoid computationally intensive operations and create this methodology a lightweight methodology. Moreover, the forward and backward access management is ensured by only permitting user access to some of the key that prohibits insiders to launch individual or coordinated attacks on the information.

2. EXISTING METHOD:

The existing, departing, and fresh joining group members will encourage be an insider threat violating information confidentiality and privacy [5]. Insider threats will encourage be more devastating because of the very fact that they are typically launched by trustworthy entities. Due to the very fact that individuals, trust, insider entities, the analysis community focus additional on outsider attackers. With all, multiple security problems will arise due to completely different users in a very cluster. We discuss a number of the problems in the following discussion. A single key shared between all cluster members can lead to the access past information to a fresh joining member. The aforementioned situation violates the confidentiality and also the principle of least privilege. Likewise, an outgoing member will access future communication. [8]Therefore, in group-shared information, the inside members may generate the difficulty of backward access management (a new user accessing past data) and forward access management (a departing user accessing future data). The easy solution of re-keying (generating a brand new key, decrypting all the information and re-encrypting with the new key) does not encourage be scalable for frequent changes within the cluster membership.[5] A separate key for each user may be a cumbersome solution. The data should be on an individual basis encrypted for each user in such a scenario. The changes within the information need the decryption of all of the copies of the users and encryption once more with the changed contents. The existing and legitimate cluster members would possibly show illegitimate behavior to govern the info. The presence of the complete symmetric key with a user permits a malicious user to turn into an insider threat. The info is decrypted, modified, and re-encrypted by a malicious business executive inside a group. Consequently, a legitimate user within the cluster might access certain unauthorized files inside the cluster. On the opposite hand, it is necessary for a user to possess a key to conduct various operations on the info. [9] The possession of the key additionally implicitly proves the legitimacy of a user to work with the data. Yet, at the same time handling each the issue regarding the key is a very important issue that must be addressed effectively.

3. PROPOSED METHOD:

In this paper, we propose a strategy named secure information sharing in Clouds that deals with the said security requirements of shared cluster information among the cloud. The secure sharing of data methodology works with 3 entities as follows:1) users; 2) a cryptographic server; and 3) the cloud. The data owner submits the information, the list of the users, and the parameters needed for generating an access control list to the cryptographic server. The cryptographic server may be a trusted third party and is responsible for key management, encryption, decryption, and access management.



[10] The cryptographic server generates the symmetric key and encrypts the information with the generated key. Later on, for every user within the group, the cryptographic server divides the key into 2 components such one part alone cannot regenerate the key. In turn, the first key is deleted through secure overwriting. One a part of the key is transmitted to the corresponding user within the cluster, whereas the opposite half is maintained by the metallic element among the access control list regarding the information file. The access control list is generated through the parameter submitted by the information owner. The encrypted information is subsequently uploaded to the cloud for storage on behalf of the user. The user who desires to access the information sends a transfer request to the cryptographic server. The cryptographic server, when authenticating the requesting user, receives the portion of the key from the user and later on downloads the information file from the cloud. The key is regenerated by operation of the user portion of the key, and the corresponding cryptographic server maintained portion for that individual user. The data is decrypted and sent back to the user. For a fresh joining member, the 2 parts of the key are generated, and the user is additional to the access control list. For an outgoing member, the record is deleted from the access control list. The outgoing member cannot decrypt the information on its own as he/she only possesses some of the key. Similarly, no frequent secret writing and re-encryption are required just in case of changes within the cluster membership. Moreover, secure data sharing of data is used by the mobile cloud computing paradigm additionally to standard cloud computing because of the fact that compute-intensive operations are performed by the cryptographic server. Our major contributions, as according during this paper, are as follows. The projected methodology ensures the confidentiality of the data on the cloud by exploitation symmetric encryption. The secure information sharing over the cloud among the cluster of users is ensured while not the elliptic curve or bilinear Diffie Hellman drawback (BDH) cryptographic re-encryption. The possession of some of the key secures the information against malicious insiders inside the cluster. The projected secure sharing of data methodology secures the information against problems with forward and backward access management that arise because of insider threats.

4. SECURE SHARING OF DATA IN THE CLOUD

The Secure sharing of data methodology maintains one cryptographic key for every of the information files. However, once encryption/decryption, the total key is not holding on and processed by any of the concerned parties. The key is divided into two constituent parts and area unit possessed by totally different entities. The subsequent are the keys that are used among Secure sharing of data in the cloud.

Symmetric Key K: K could be a random secret generated by the cryptographic server for every of the info files. The length of K in Secure sharing of data is256 bits, as is suggested by most of the standards concerning key length for symmetric key algorithms (SKAs). However, the length of the key may be altered according to the requirementsof the underlying SKA. K is obtained in a two-step method. In the opening, a random variety R of length 256 bits are generated such R = 256. Within the next step, R is passed through a hash function that might be any hash operate with a256-bit output. In our case, we used secure hash algorithm256 (SHA-256). The second step fully randomizes the initial user-derived random range R. The output of the hash function is termed as K and is employed in symmetric key encoding [e.g., The Advanced encryption normal (AES)] for securing the information. The cryptographic key share and user key share also maintains in this method. The Cryptographic key share is generated for every user in the group, the cryptographic providesKI, whenever cryptographic received the encryption and decryption request by the user.

4.1Secure sharing of data in the cloud design:

Upload process:

Whenever a need to share information among the group arises, the owner of the file sends the encryption request to the cryptographic server. The request is among the file and a list of users that are to be granted access to the file. L also contains the access rights for every of the users. The users might have Read-only and/or READ–WRITE access to the file. Other parameters are also set to enforce fine-grained access management over the information. L is used to get the Access control list for the information by the Cryptographic server. L is shipped to the cryptographic server given that the information is to be shared with a new proposed cluster. If the cluster already exists, the encryption request won\'t contain L; rather, the cluster ID of the present group is going to be sent. The Cryptographic server, when receiving the encryption request for the file, generates the Access control list from

the list and creates a group of the users. The Access control list is individually maintained for each file. The Access control list contains data concerning the file such as its unique ID, size, owner ID, the list of the user IDs with whom the file is being shared, and different data. If the group already existed, only the Access control list for the file is formed.

Next, the cryptographic server generates K and encrypts the file with an acceptablesymmetric block cipher (we have used the AES for encryption type purposes). The result is an encrypted file. After, the cryptographic server generates Ki and Ki for each user and deletes K by secure overwriting. Secure overwriting may be an idea in which the bits within the memory are constantly flipped to create certain that a memory cell never grips a charge for enough period for it to be remembered and recovered. The Ki for every user is inserted into the Access control list for later use. To guard the integrity of the file, the Cryptographic Server conjointly computes the hash-based message authentication code (HMAC) signature of each encrypted file. The same procedure for the HMAC secret is adopted. However, the HMAC secret is kept by the cryptography server only. The encrypted information, the cluster ID (in the case of a recently generated group), and therefore the Ki for the owner are sent to the requesting information owner. The cluster ID and therefore the Ki for the remainder of the cluster users are directly sent to them over a secure communicating. The general public key of the group users is also used to transmit the user portion of the key. We have used the general public keys of the users to transmit the key parts. The user, when receiving a C, uploads it to the cloud. K is deleted via secure overwriting from the cryptographic key when the encryption method. Algorithm one shows the key generation and encryption methodin the Cryptographic server. It is noteworthy that the key generation method is executed once once the cluster is initiated and therefore the initial file is submitted in secret writing. Moreover, a recently joining member also activates the key generation, however, just for the new member. It is important to notice that, when the encryption of the information on the Cryptographic server, the uploading of the file to the cloud is handled in two potential ways in which. Within the initial possibility, the encrypted information is sent to the user who uploads it to the cloud, as explained earlier during this section. Within the second possibility, the cryptographic server is delegated the authority to transfer the file to the cloud on behalf of the user. We have used the second possibility in our implementation.

Download process:

The approved user sends a download request to the cryptographic server or downloads the encrypted file from the cloud and sends the decryption request to the cryptographic server. The cloud verifies the authorization of the user through a regionally maintained Access control list. The decryption request is accompanied by the user portion of the key, i.e., Ki, alongside different authentication credentials. The cryptographic server computes K by applying an XOR operation over Ki and therefore the corresponding ki from the Access control list. As each of the users corresponds to a unique combine of Ki and Ki, none of the users will use different users Ki to masquerade identity. Subsequently, the cryptographic servers return with the decryption method after collateral the integrity of the file. If the proper Ki is received by the cryptographic server, the results are successful decryption process; otherwise, the decryption can fail. After successful decryption, the file is sent to the requesting user through a secure communicating that would be Secure Sockets Layer (SSL) or net Protocol Security (IPSec) channels. K is deleted via secure overwriting from the cryptographic server when decryption. The users are genuine before the request process according to standard procedures. The algorithmic program a pair of presents the decryption method.

5. ANALYSIS OF SECURE SHARING OF DATA IN THE CLOUDS

The Secure sharing of data methodology is proposed to produce the following services to the outsourced data: confidentiality, secure information sharing among the group, secure information from unauthorized access by valid insiders within the group and forward and backward access management to counter insiders and departing cluster users. The following discussion in short description, however the same services are achieved. We do not consider the cloud to be a secure and trustful entity in the context of Secure sharing of data in the cloud.

Multitenancy, virtualization, and a shared pool of resources might create several forms of insider and other threats to the information. Moreover, the cloud may additionally retain copies of the file even once it is requested for deletion.

For secure information sharing, Secure sharing of data does not utilize the idea of re-encrypted with multiple keys. The encryption is completed with one symmetrical key. However, the approved users are granted access on the idea of possession of the key share and the typical authentication and authorization development. The Access control list, lists the approved users with their credentials and corresponding cryptographic server key shares. After authentication, the user's share of the key is used, at the side of the cryptographic server share, to get K. Because the user share is just possessed by a valid user, only a valid user will result in successful encryption/decryption of the data. The division and dispersal of the key also helps counter the insider malicious users among the cluster. The Access control list is singly maintained for every cluster file. Therefore, a valid cluster user cannot access the cluster file that\'s not shared with him/her. An attempt to access an unauthorized file is additionally blocked by the actual fact that the user will not have the key share of that file. Moreover, the Access control list of the unauthorized files will not contain any record for the malicious user. Moreover, the absence of the complete key with the user and therefore the Access control list put together ensures the forward and backward access management for the information.

Most of the information forwarding schemes are dependent on the El-Gamal cryptosystem and bilinear pairing [7]. The aforementioned schemes need the re-encryption of the info each time the access to the information is requested by any user however the owner. The El-Gamal cryptosystem is computationally intensive. Moreover, re-encryption at every access adds to the overhead. The secure sharing of data methodology utilizes symmetric cryptography, and the access to multiple users are achieved through key management, as explained within the preceding section. Therefore, the overhead of the secure sharing of data methodology is fairly less as compared with the standard El-Gamal based re-encryption systems.

6. CONCLUSION

We planned the secure sharing of data methodology that could be a cloud storage security theme for cluster knowledge. The planned methodology provides knowledge, confidentiality, secure knowledge sharing without re-encryption, access management, for malicious insiders, and forward and backward access management. Moreover, the secure sharing of data methodology provides assured deletion by deleting the parameters required to decipher a file. The coding and decoding functionalities are performed at the cryptographic server that is a trustworthy third party within the secure sharing of data methodology. The planned methodology can be also employed to mobile cloud computing because of the fact that compute-intensive tasks are performed at the cryptographic server. The results unconcealed that the Secure sharing of data methodology will be much utilized in the cloud for secure knowledge sharing among the group. In the future, the planned methodology is extended by limiting the trust level within the cryptographic server. This can any enhance the system to handle insider threats. Moreover, the response of the methodology with varied key sizes is evaluated.

REFERENCES

[1] A. Abbas and S. U. Khan, "A review on the State-of-the-art privacy preserving approaches in e-health clouds," IEEE J. Biomed. Health Informat., vol. 18, no. 1, pp. 1431–1441, Jul. 2014.

[2] K. Alhamazani et al., "An overview of the commercial cloud monitoring tools: Research dimensions, design issues, state-of-the-art," Computing, DOI: 10.1007/s00607-014-0398-5, 2014, to be published.

[3] L. Xu, X. Wu, and X. Zhang, "CL-PRE: A certificateless proxy re-encryption scheme for secure data sharing with public cloud," in Proc. 7th ACM Symp. Inf. , Comput. Commun. Security, 2012, pp. 87–88.

[4] S. Seo, M. Nabeel, X. Ding, and E. Bertino, "An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds," IEEE Trans. Knowl. Data Eng., vol. 26, no. 9, pp. 2107–2119, Sep. 2013.

[5] A. N. Khan, M. M. Kiah, S. A. Madani, M. Ali, and S. Shamshir-band, "Incremental proxy re-encryption scheme for mobile cloud computing environment," J. Supercomput., vol. 68, no. 2, pp. 624–651, May 2014.

[6] A. N. Khan, M. L. M. Kiah, S. U. Khan, and S. A. Madani, "Towards secure mobile cloud computing: A survey," Future Gen. Comput. Syst., vol. 29, no. 5, pp. 1278–1299, Jul. 2013.

[7] L. Wei, H. Zhu, Z. Cao, Y. Chen, and A. V. Vasilakos, "Security and privacy for storage and computation in cloud computing," Inf. Sci., vol. 258, pp. 371–386, Feb. 2014.

[8] Cloud security Alliance, "Security guidelines for critical areas of focus in cloud computing v3.0," 2011.

[9] D. Chen et al., "Fast and scalable multi-way analysis of massive neural data," IEEE Trans. Comput., DOI: 10.1109/TC.2013.2295806, 2014, to be published.

[10] Y. Chen, J. D. Tygar, andW. Tzeng, "Secure group key management using uni-directional proxy re-encryption schemes," in Proc. IEEE INFOCOM,pp. 1952–1960.