Scientific Journal Impact Factor (SJIF): 1.711

e-ISSN: 2349-9745
p-ISSN: 2393-8161

International Journal of Modern Trends in Engineering and Research

www.ijmter.com

Improve HLA based Encryption Process using fixed Size Aggregate Key generation

Shruti Batham[1], Umesh Lilhore[2], SiniShibu[3]
[1, 2, 3] CSE Department, NIIST College Bhopal

Abstract—Cloud computing is an innovative idea for IT industries which provides several services to users. In cloud computing secure authentication and data integrity of data is a major challenge, due to internal and external threats. For improvement in data security over cloud, various techniques are used.MAC based authentication is one of them, which suffers from undesirable systematic demerits which have bounded usage and not secure verification, which may pose additional online load to users, in a public auditing setting. Reliable and secure auditing are also challenging in cloud. In Cloud auditing existing audit systems are based on aggregate key HLA algorithm. This algorithm is based on variable sizes, different aggregate key generation, which encounters with security issues at decryption level. Current Scheme generates a high length of key decryption that encounters with problem of space complexity. To overcome these issues, We can improve HLA algorithm by improve aggregate key generation, based on fixed key size. This algorithm generates constant aggregate key which will overcomes problem of sharing of keys, security issues and space complexity.

Keywords –Cloud Computing, TPA, Aggregate Key generation, HLA, Secure Hash Algorithim-1.

I. INTRODUCTION

Cloud computing is a new and an innovative idea of 21st Century for IT industries. The term cloud computing is referred as “The Cloud” which used as a “Metaphor” for the “internet” so cloud computing is a “type of internet based computing”. The purpose of the cloud computing is dynamically deliver the computing resources and capabilities as a services over the web. It is a new technology of computing in which dynamically scalable and often virtualized resources are provide as a service over the internet [1].Cloud computing involved different hosted services over the internet which are IaaS (Infrastructure-as-a-service), PaaS (Platform-as-a-service), and SaaS (Software-as-a-service) .IaaS service provide the infrastructure like memory, space, storage etc to users. Saas service provides different application rather installing to its own system. PaaS service provides cloud application to developer and responsible for virtualization of resources and makes it as a single layer. Cloud deployment models are-

Public cloud: A public cloud is one standard of cloud computing, in which a cloud service provider provide a virtual environment, make a pool shared resources, such as applications and storage, offered to the general public over the web. Public cloud services are offered to users as pay-per-usage.

Private cloud: Organizations choose to build their private cloud as to keep the strategic, operation and other reasons to themselves and they feel more secure to do it.

Hybrid model: It consists of multiple service providers. It provides the services of both public and private cloud. It is used by organization when they need both private and public clouds both.

Cloud computing is adapted by many business organization, industries and publicly in the recent years for using its application, resources and data storage for that security is a major concerns regarding cloud which is not provided by cloud service provider as it is a separate administrative entities. This problem is not properly addressed in the successful deployment of the cloud architecture. As user's no longer physically possess the storage of their data, traditionally cryptography technique is not directly adopted to ensure the data security protection. Sometimes the cloud service provider may hide the data corruption for maintain their reputation in the market. In particular, simply downloading the data for checking its integrity is not a solution because as it provide a extra burden in network and expensive also it is not sufficient to detect the data corruption. To avoid this problem we proposed an efficient solution in this paper which is TPA (third party auditor) to audit the user's outsourced data when needed.

1.2 Third party auditor fully ensures the user's data integrity, availability and save the cloud computation resources as well as online burden. TPA has expertise and capabilities that cloud users do not check the integrity of user's data periodically which are stored in cloud on the behalf of the user's. TPA provides an affordable way and assurance users for the security of the data which are stored in the cloud. TPA is just for verifying the security of the stored data. Exploiting data encryption before outsourcing is one way to mitigate this privacy concern but this is just a complementary to the privacy preserving public auditing to be proposed in this paper. But without proper auditing protocol it is not helpful for preserving the security still there is some leakage to an external parties its just reduce the key management. Therefore for enabling a privacy preserving third party auditing protocol is independent to data encryption, is the problem we are tackle in this paper. Our work is to support the privacy preserving public auditing in cloud computing, focus on data storage. Also for TPA is to tedious and cumbersome while verifying the security of data one by one so in our paper it enable to TPA verifying the security of user's data stored in a cloud in a multiple or in a batch manner auditing. To solve this problem our work is helpful by providing a technique which is based on constant aggregate key, and SHA-2. TPA performs auditing without the need of a local copy of data which reduces the computational and communication overhead.

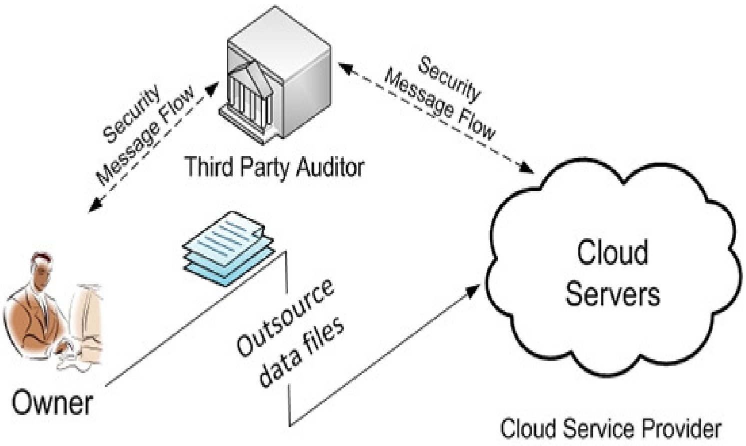

Figure 1. Cloud computing

II. LITERATURE SURVEY

Cong wang, Qian Wang , Kui Ren, Wenjing Lou [1] " Privacy Preserving Public Auditing for secure cloud data storage" using cloud storage , users can remotely store their data and enjoy the on demand high quality applications and services from a shred pool of configurable computing resources, without the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the outsourced data makes the data integrity protection in cloud computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and worry-free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. In this they propose a secure cloud storage system supporting privacy-preserving public auditing. Further TPA perform audits for multiple users simultaneously and efficiently.

Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jianying Zhou and Robert H. Deng[2] "Key-Aggregate Cryptosystem for scalable data sharing in cloud storage" Data sharing is an important functionality in cloud storage. They describe a public key cryptosystems which produce constant size ciphertexts such that efficient delegation of decryption rights for any set of ciphertexts are possible. They novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but encompassing the power of all the keys being aggregated. In other words, the secret key holder can release a constant size aggregate key for flexible choices of ciphertext set in cloud storage, but the other encrypted files outside the set remain confidential.

Ms. A. Christy Sharmila, Mr. T. Tressiline [3] "Cloud audit service by using KAC for data storage security" Cloud Computing is recognized as an alternative to traditional information technology due to intrinsic resource sharing and low maintenance characteristics. Enterprises usually store data in internal storage and install firewalls to protect against intruders to access the data. With proven security relied on number theoretic assumptions is more desirable, whenever the user is not perfectly happy with trusting the security of the virtual machine or the honesty of the technical staff. The challenging problem is how to effectively share encrypted data.

Akkala. Saibabu, T. Satyanarayan Murthy [4] Security is achieved by signing the data block before sending to the cloud. Sighning is performed using sha-1 algorithm which is more secure compared of data, allowing a third party auditor on behalf of the cloud user to verify the integrity of the data stored in the cloud. By utilizing public key based homomorphic authenticator with random masking privacy preserving public auditing can be achieved. The technique of bilinear aggregate signature is used to achieved batch auditing.

III. PROBLEM FORMULATION

Limitation of Exiting Aggregate key HLA algorithm in Batch Processing of data:

[1] HLA based on homomorphic linear authentication which not privacy preserving because in HLA linear combination of block may potentially reveal user data information to TPA.

[2] The High Level Architecture (HLA) and its Run-Time Interface (RTI) do not define support of mandatory access controls (MACs) or discretionary access controls (DACs) required providing necessary protection levels.

[3] Different size of aggregate key generation which may cause problem while decryption level Current Scheme may generate a high length or high size key-we cannot expect large storage for decryption keys in the resource-constraint devices like smart phones, smart cards or wireless sensor nodes. Especially, these secret keys are usually stored in the tamper-proof memory, which is relatively expensive.

[4] At the time of decryption it may take long while performing decryption with long length key.

[5] The TPA also has to maintain and update state between audits i.e., keep track on the revealed MAC keys. Considering the potentially large no. of audit delegation from multiple users maintaining such states for TPA can be difficult.

IV. PROPOSED METHODOLOGY

In modern cryptography, a fundamental problem we often study is about leveraging the secrecy of a small piece of knowledge into the ability to perform cryptographic functions (e.g. encryption, authentication) multiple times. In this paper, we study how to make a decryption key more powerful in the sense that it allows decryption of multiple cipher texts, without increasing its size. We solve this problem by introducing a special type of public-key encryption which we call key-aggregate cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. That means the cipher texts are further categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a dynamic class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes.

Algorithm- HLA algorithm **for fixed size aggregate Key generation**

Input- 1, n, i , m, h, S.

// 1- security level parameter, n- cipher text classes n.

// i- index which denote the cipher text class, m- message

// h- hash value, S- set of indices corresponding to different classes

Output- C, H, KS, Alert message True/ False.

//C- Cipher text, H- Hash value, KS- Aggregate key.

Step1: Setup (1_; n):

1.1 Executed by data owner to setup an account on an un trusted server.

1.2 Choose security level parameter 1 and cipher text classes n.

1.3 By step 1.2 Public system parameter param is generated.

1.4 KeyGen

1.4.1 Executed by the data owner to randomly generate a (pk; msk).

// pk- public key, msk-master-secret key pair.

1.5 **Encrypt (pk; i; m):**

1.5.1 For encrypting the data, apply pk, i , m on the plain text.

1.5.2 By step 1.5.1. cipher text C is obtained.

1.5.3 Hashing operation is applied on C.

1.5.4 C and H are to be stored in cloud

Step 2: **Extract (msk; S):**

2.1 executed by the data owner for delegating the decrypting power for a certain set of cipher text classes.

2.2 msk and S are applied on cipher text classes which generate aggregate key KS of constant size.

Step 3: **Decrypt (KS; S; i; C)**

3.1 executed by a delegate who received an aggregate key KS generated by Extract.

3.2 If user is authenticate than it apply the keys KS, S, i on C.

3.3 from the above step original message is obtained.

3.4 else user is unable to obtain message and error message is occurred.

Step 5: **TPA verify the integrity of data.**

5.1 TPA select a random block of cipher text message.

5.2 Send a challenge to CSP

5.3 CSP evaluates the challenge.

5.4 TPA evaluate the hash value on block of cipher text.

5.4 TPA generates the alert message True/False. After matching the hash value evaluated by CSP and TPA.

This algorithm Generates Constant Aggregate key which overcomes problem of sharing the key or using at decryption level.

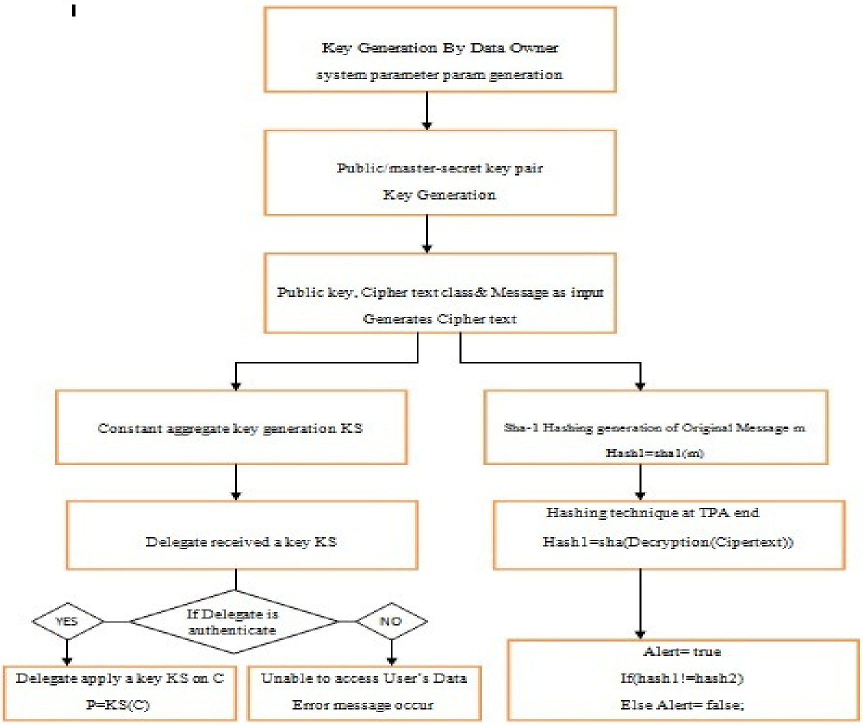
Key Generation By Data Owner
system parameter param generation
Public/master-secret key pair
Key Generation
Public key, Cipher text class& Message as input
Generates Cipher text
Constant aggregate key generation KS
Sha-1 Hashing generation of Original Message m
Hash1=sha1(m)
Delegate received a key KS
Hashing technique at TPA end
Hash1=sha(Decryption(Cipertext))
YES
If Delegate is authenticate
NO
Delegate apply a key KS on C
P=KS(C)
Unable to access User's Data
Error message occur
Alert= true
If(hash1!=hash2)
Else Alert= false;

Figure 2. Flow of Algorithm

V. Expected outcome

To ensure the data integrity and authentication we enhance the HLA algorithm by improve aggregate key generation, based on fixed key size. This algorithm generates constant aggregate key which will overcomes the problem of sharing of keys, security issues and space complexity. After simulation our expected outcomes are:

i) Decryption key is of constant size.

ii) The number of cipher text classes is constant which are increased dynamically.

iii) Multi user's data are audit in batch manner.

VI. CONCLUSION

Cloud Computing is gaining remarkable popularity in the recent years for its benefits in terms of flexibility, scalability, reliability and cost effectiveness. Despite all the promises however, Cloud Computing has one problem: Security, we studied the problems of data security in cloud data storage, which is essentially a distributed storage system. An effective and flexible distributed scheme is proposed to ensure the login into the cloud server & then checking correctness of user's data in the cloud servers. By detailed security and performance analysis, we show that our scheme is highly efficient in recovering the singleton losses almost immediately and recovers from busty data losses. In future this may be implemented in the field of Intrusion detection technique.

REFERENCES

[1] Cong Wang, S.M.Chow, Qian Wang, Kui Ren, Wenjing Lou “Privacy-preserving public auditing for secure cloud storage” IEEE transactions on computers vol:62 no:2 year 2013.

[2] Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jinaying Zhou and Robert H. Deng “Key aggregate cryptosystem for scalable data sharing in cloud storage” IEEE transactions on parallel and distributed systems, vol:25 issue 2, year 2014

[3] Ms. A. Christy Sharmila, Mr, T. Darney Tressiline “Cloud audit service outsourcing by using KAC for data storage security” International journal of engineering research and applications (IJERA)ISSN:2248-9622 International conference on humming bird 01st march 2014).

[4] Akkala. Saibabu, T. Satyanarayan Murthy “Security provision in publicly auditable secure cloud data storage of computer services using sha-1 algorithm” et al, International journal of computer science and information technologies Vol.3(3), 2012.

[5] R. Vanitha “An Aggregate Key Based Cryptosystem for Secure Data Sharing in Cloud Computing” *et al,* International Journal of Computer Science and Mobile Computing, Vol.3 Issue.4, April- 2014.

[6] B. Wang, S. S. M. Chow, M. Li, and H. Li, “Storing Shared Data on the Cloud via Security-Mediator,” in International Conference on Distributed Computing Systems - ICDCS 2013. IEEE, 2013.

[7] B. Wang, S. S. M. Chow, M. Li, and H. Li, “Storing Shared Data on the Cloud via Security-Mediator,” in International Conference on Distributed Computing Systems - ICDCS 2013. IEEE, 2013.

[8] S. Yu, C. Wang, K. Ren, and W. Lou, “Achieving secure, scalable, and fine-grained access control in cloud computing,” in *Proc. of IEEE INFOCOM'10*, San Diego, CA, USA, March 2010.

[9] W.-G. Tzeng, “A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy,” IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 14, no. 1, pp. 182–188, 2002.

[10] T. Okamoto and K. Takashima, “Achieving Short Ciphertexts or Short Secret-Keys for Adaptively Secure General Inner-Product Encryption,” in Cryptology and Network Security (CANS '11), 2011.

[11] Y. Sun and K. J. R. Liu, “Scalable Hierarchical Access Control in Secure Group Communications,” in Proceedings of the 23th IEEE International Conference on Computer Communications (INFOCOM '04). IEEE, 2004.

[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained access control in cloud computing," in *Proc. of IEEE INFOCOM'10*, San Diego, CA, USA, March 2010.

[13] G. Ateniese, A. D. Santis, A. L. Ferrara, and B. Masucci, "Provably-Secure Time-Bound Hierarchical Key Assignment Schemes," J. Cryptology, vol. 25, no. 2, pp. 243–270, 2012.

[14] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," in Proceedings of Advances in Cryptology - EUROCRYPT '05, ser. LNCS, vol. 3494. Springer, 2005, pp. 457–473.

[15] D. Boneh and M. K. Franklin, "Identity-Based Encryption from the Weil Pairing," in Proceedings of Advances in Cryptology - CRYPTO '01, ser. LNCS, vol. 2139. Springer, 2001, pp. 213–229