Indian Journal of Science and Technology, Vol 9(44), DOI: 10.17485/ijst/2016/v9i44/90138, November 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

An Efficient Framework for Providing Secured Transaction of Data in Cloud Environment

T. Brindha[1]* and R. S. Shaji[2]

[1]Department of Computer Science and Engineering, Noorul Islam University, Kumaracoil - 629180, Tamil Nadu, India; brindha@niuniv.com.
[2]Department of Information Technology, Noorul Islam University, Kumaracoil - 629180, Tamil Nadu, India; shajiswaram@niuniv.com

Abstract

Background/Objectives: The aim our work is to achieve transactional security for users across cloud servers. **Method/Statistical Analysis:** A framework called Conditional Source Encryption based Data Transactional Security (CSEDTS) is developed in cloud environment. The transaction request sent to cloud data storage is evaluated as conditional attributes. The obtained conditional attributes are encrypted with the help of Conditional Source Encryption method. Then, mapping function is applied for conditional attributes by using unique secured identity number and then these conditional attributes gets decrypted. **Findings:** Experiments are conducted and the performance analysis shows that the transactional security rate on data layer and the true positive rate are improved. **Applications/Improvements:** This technique increases security of data that can be applied to access a wide range of resources and applications through web service interface. In future, this can be improved by reducing the transaction time.

Keywords: Conditional Attribute, Cloud Server, Data Storage, Encryption, Security, Transaction

1. Introduction

Cloud computing is a deep imagined perception of computing[1] as a service, where the clients can place their data remotely in the cloud to make use of applications and services based on their interest from a common pool of computing resources. Cloud is wrapped around advanced technologies like server virtualization which provides benefits of multitenancy and scale economy with the aim of lowering the charge of adopting the IT resources. Since the cloud permits the users to store their huge volume of data into the cloud, the user's burden of maintaining the data in the local machines has been decreased.

Although the advantages are obvious, the cloud storage introduces the fear of data accuracy for the users. The security issues on cloud are increased with respect to privacy, integrity, availability etc. Moreover, it provides unusual and challenging threats[2] to security for the data being outsourced. It also brings new and challenging security threats to the outsourced data. Since the data outsourced to cloud are maintained by cloud service providers[3,4], the data owners control over the data is ultimately dropped out. Because of its multi-tenancy feature, outsourcing of private data, demanding applications and infrastructure onto the cloud, more security and privacy issues has been raised in cloud computing. Concerns on how to maintain the security and privacy in the emerging cloud environment arises in major organizations and by individuals[4]. Also, the companies are having strict constraints on outsourcing their private data and demanding applications on public clouds[5,6].

Even though cloud computing provides on-demand accessing of computing resources[7], the lack of security forms a major issue. The widespread usage of cloud computing resources will be limited unless the cloud users begin to entirely trust the cloud providers. The legal and technical facets increase the issue of cloud security. Since the cloud infrastructure groups the various different

*Author for correspondence

services and software developed by various development teams without sharing approach forms a major challenge in ensuring cloud security[8].

Data becomes a great concern when it is outsourced to cloud[9,10]. Hence, the most active domain of research in cloud computing is concentrating on the security and privacy of cloud data. Preventing the leakage of data and protecting the sensitive data becomes essential for most of the companies moving on to cloud.

2. Related Works

Cloud computing is a one of the effective computing methods as compared to the conventional form of desktop computing. Today, this new technology has received great attention by researchers and organizations. The Remote Data Auditing (RDA)[11] method is introduced to provide remote data storage in single cloud server domain with aim of improving the retrievable rate. Cloud data transaction was performed in a capable manner by different users at various access levels, but it fails to provide optimal security framework. Shield[12] was designed with the objective of improving security using Merkle Hash Tree without the need of modifying the file system. However, the cloud data storage technique in Shield has not concentrated on maximizing the security on performing the transactions over cloud servers. Therefore, both above methods mentioned have lack of security.

Multiple data owners are considered in which the entire system is divided into numerous domains[13]. The data is encrypted using AES technique and then for the purpose of broadcasting, the AES key is encrypted using Attribute Based Broadcast Encryption (ABBE) technique which uses the limitless size on attributes. But it creates complexity in providing immediate revocation in case of multiple authorities being online. An architecture based on cloud computing[14] is used to secure the data and retain the sensitive information regarding the location of user data. The information regarding the location of data is identified using Global Positioning System (GPS). The limitation of this technique is that it works only in GPS enabled systems.

Optimal Integrity Policy method[15] is used for data security in which private keys are generated based on AND, EXOR and hashing operations and then integrity of data is verified by using MAC process. Even though, the security is enhanced in this process, the decryption is very slow for the devices. The confidentiality is provided in[16], where instead of decryption, multiplicative property based on paillier algorithm and additive property is applied for data sets. This technique is used for improving the security but it does not concentrate on integrity checking. The confidentiality of data is enhanced by using MONcrypt mechanism[17] in which the data to be outsourced are obfuscated which converts the plain text into ASCII characters. The size of the data outsourced to be stored in cloud is reduced. An Instance Communication Channel Key Organizer (ICCKO)[18] model is presented which reduces the leakage of data in cloud thereby increasing the security of data.

Cloud computing has evolved as most significant pattern for the design and analysis of virtual environment over Internet. In[19], an efficient resource allocation scheme was designed with basic quality of service using win-win effect and incentive compatibility obtained. Optimization of cloud task processing was developed[20] to improve the execution performance through composite cloud service system. Hybrid Particle Swarm Optimization[21] for grid computing was designed to reduce the overall completion time. This in turn improves transaction time over the three methods. To improve data storage and security in cloud, an efficient data storage auditing mechanism was introduced[22]. However, higher level of security was not guaranteed.

The privacy-preserving public auditing mechanism were presented[23] for data storage protection using Advanced Encryption Standards (AES) encryption algorithm in cloud computing. Though the computational time is improved, but the privacy is not high. In this work, an efficient framework called Conditional Source Encryption based Data Transactional Security (CSEDTS) is designed to ensure transactional security on performing the transactions over cloud servers.

The rest of the paper is organized as follows. Section 2 describes our CSEDTS framework. Section 3 explains the experimental results. Section 4 evaluates the performance of CSEDTS model by simulation. Section 5 concludes the paper.

3. Secure Data Transaction

The detailed structure of Conditional Source Encryption based Data Transactional Security (CSEDTS) framework is constructed. The framework provides high secure transactions across different conditional attributes. Figure 1 shows data transactional security mechanism in cloud.

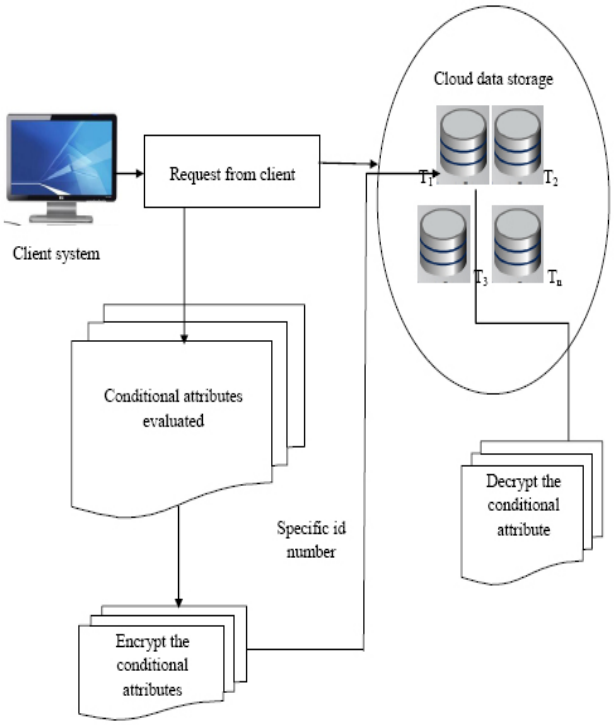

Figure 1. Architecture diagram of CSE-CDTS framework.

The client sends requests to the cloud data storage system for the purpose of transactional processing. The proposed framework is concentrated on encrypting the conditional attribute from the source root systems, to improve the security level measure. The client system sent request to cloud data storage for transactional processing. At first, the conditional attribute is evaluated by Bilinear Mapping Transformation function. Bilinear function performs one to one mapping to improve the transaction processing security. Finally, the conditional requests are updated to the decrypted conditional attribute from cloud data storage using CSEDTS framework. The concept used in CSEDTS framework increases the stochastic nature of the particle and attain maximum result with better solution.

3.1 Conditional Attribute Encryption

The first step involved is the design of Conditional Source Encryption based Data Transactional Security (CSEDTS) framework for performing cloud data transactions. The data requested from the client is evaluated as conditional attributes. The attributes are evaluated as,

$$CA = \{A, T, D\} \rightarrow S_g \quad (1)$$

From equation (1), the conditional attribute evaluation is obtained by selecting the variables attributes 'A' and request to the attributes to be transacted through '*T*' from the total data storage in cloud 'D'. 'S_g' forms the syntactic generator for the request query from the client side.

The client initiates the transaction request *T* through the cloud applications. The transaction request performed with conditional attributes to be encrypted for high secure transactional requirement. The conditional attribute information is encoded on the data layer using the bilinear mapping transformation. The data layers are transformed into a set of encrypted request from the clients used to improve the security with cipher type of message request. Then, client and server accept the request message used for transaction processing. The mapping function preserves the separate attributes from the clients to perform high effective transactions.

The algorithm for conditional attribute encryption with cipher specific id is as follows:

Algorithm 1: Conditional attribute encryption algorithm

Begin

- Split simple attribute request into a number of blocks.
- Measure conditional attribute splitting as $CA1 + CA2$.
- Generate cipher id with pseudo random number using $\frac{CA1I+C_a}{CA2I+C_b}$.
- Generate specific key id generation based on cipher text using set Key () function.
- Converted cipher text and Specific id is sent to cloud storage server.
- Repeat step 1 to 4 for entire request from client systems.

End

3.2 Mapping Process for Transaction Processing

The mapping function designed for improving transaction efficiency. It is used to perform the mapping process for transaction processing security in CSEDTS framework. The cloud service provider obtains the specific key $k_{i1}, k_{i2}, \ldots . k_{in}$ and cipher text of user's request obtained from conditional attribute encryption algorithm. The CSP manages the data block location to the corresponding data owners. Finally, mapping algorithm performs

the linear mapping with prestored keys in cloud service provider using CSEDTS framework.

3.3 Conditional Attributes Decryption

Finally, the conditional attribute decryption is significantly performed in the design of CSEDTS framework. In the server side, the conditional attribute is decrypted in order to provide transaction time for whole transaction process in CSEDTS framework. The decryption procedure on the cloud server side is employed to obtain the specific user id with the encrypted conditional attribute. The decryption procedure on the cloud server side is formalized as given below,

The conditional attribute decryption algorithm is described as,

Algorithm 2. Conditional Attribute Decryption

Begin

Input: Fetch the Encrypted conditional attribute with specific id

- Translate cipher message into decimal factor
- Find exact bilinear mapping as $\frac{C_a - C_{id}\, C_b}{C_{id}\, CA2 - CA1}$ where'$CA1$ 'and '$CA2$ 'represents the conditional attributes for original specific id using two constants 'C_a and 'C_b 'with 'C_{id} 'representing the cipher specific id.
- Original Specific id makes the original request message from the binary form.
- Repeat the step 1 to 3 for the entire specific request id.

End

4. Experimental Evaluation

The Conditional Source Encryption based Data Transactional Security framework uses the CloudSim simulator to work under the simulation environment. The experimental work is carried out for evaluating the security level on the transactions between different cloud environments. With the simulation speed is 8 GB of RAM and 1 TB of storage space. Amazon Access Samples dataset information is used on the transaction processing between clients and server systems. The Amazon Access Samples dataset includes four categories of attributes including Person_Attribute, Resource_ID, Group_ID and System_Support_ID.

5. Discussion

The proposed CSEDTS framework is compared against with the Remote Data Auditing (RDA)[14] method and Stackable Secure Storage System (Shield)[13].

5.1 Impact of Transactional Security Rate

The impact of transactional security rate for CSEDTS framework is elaborated and comparison is made with two other methods RDA and Shield respectively. Table 1 represents the security level obtained using CloudSim simulator.

Figure 2 explains the transactional security rate obtained with respect to number of request. The transactional security rate using the proposed CSEDTS framework is higher when compared to two other existing methods namely, Remote Data Auditing (RDA) [14] method and Stackable Secure Storage System (Shield) [13]. This is because of the application of Conditional Attribute Encryption and Conditional Attribute Decryption algorithm that efficiently uses the mapping function therefore increased the transactional security rate on data layer. In addition, the cipher specific id generation is obtained through bilinear function based on different client requests. This in turn helps to increase transactional security rate by 7.5 % when compared with the RDA[14] and 11.8 % when compared to Shield[13] respectively.

5.2 Impact of True Positive Rate

The impact of true positive rate for CSEDTS framework is presented in Table 2.

Figure 3 explains the true positive rate obtained with respect to the number of attributes. The true positive rate is higher for the proposed CSEDTS framework when compared to two other existing methods namely, Remote Data Auditing (RDA)[14] method and Stackable Secure Storage System (Shield)[13]. This in turn increases the true positive rate when compared with the RDA[14] and Shield[13] respectively.

Table 1. Tabulation for transactional security rate

Methods	Transactional security rate (%)
CSEDTS	93
RDA	86
Shield	82

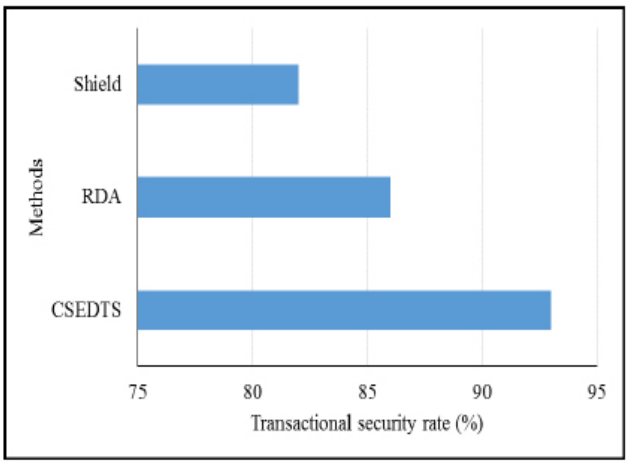

Figure 2. Measure of transactional security rate.

Table 2. Tabulation for true positive rate

No. of attributes	True positive rate (%)		
	CSEDTS	RDA	Shield
3	62.35	54.38	58.21
6	68.65	56.62	61.35
9	71.24	58.23	63.29
12	73.42	60.24	64.37
15	75.14	63.25	67.12
18	76.89	67.68	69.39
21	81.36	68.74	71.28

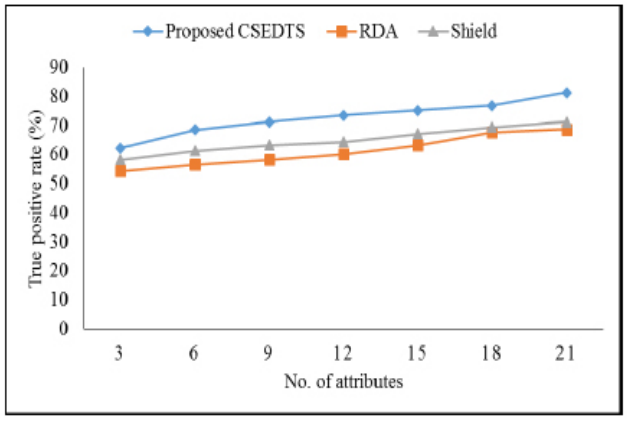

Figure 3. Measure of true positive rate.

6. Conclusion

In this work, an effective framework called Conditional Source Encryption Data Transactional Security (CSEDTS) is presented. This framework increases the performance of transaction security for different user conditional request using mapping function using the specific id number. The transaction request is evaluated in an efficient manner and as a result, the proposed framework improves the transactional security rate on data layer for each client requests resulting in improved transaction processing. The results show that the CSEDTS framework offers better performance when compared to the state-of-the-art methods.

7. References

1. Vairagade RS, Vairagade NA. Cloud computing data storage and security enhancement. IJARCET. 2012 Aug; 1(6):145-9.
2. Purushothaman D, Abburu S. An approach for data storage security in cloud computing. International Journal of Computer Science Issues. 2012 Mar; 9(2):100-5.
3. Bhisikar P, Sahu A. Security in data storage and transmission in cloud computing. International Journal of Advanced Research in Computer Science and Software Engineering. 2013 Mar; 3(3):410-5.
4. Divya SV, Shaji RS, Venkadesh P. A comprehensive data forwarding technique under cloud with dynamic notification. Research Journal of Applied Sciences, Engineering and Technology. 2013 Jul; 7(14):2946-53.
5. Mohamed SPM, Shaji RS. An efficient framework to handle integrated VM workloads in heterogeneous cloud infrastructure. Soft Computing. 2016 Jan; 1-10.
6. Sugumar R, Imam SBS. Symmetric encryption algorithm to secure outsourced data in public cloud storage. Indian Journal of Science and Technology. 2015 Sep; 8(23):1-5.
7. Pandey A, Gond S. Secure communication over cloud computing network using OTP (one time transaction) method. Int J Computer Technology and Applications. 2014 Sep; 5(5):1707-10.
8. Vamsikrishna V, Boominathan P. Authorization based secure data transaction in cloud computing. IJETT. 2014 May; 11(9):446-9.
9. Youssef AE, Alageel M. A framework for secure cloud computing. International Journal of Computer Science Issues. 2012 Jul; 9(4):487-500.
10. Shaikha R, Sasikumar M. Data classification for achieving security in cloud computing. Procedia Computer Science. 2015; 45:493–8.
11. Sookhak M, Talebian H, Ahmed E, Gani A, Khan MK. A review on remote data auditing in single cloud server: Taxonomy and open issues. Journal of Network and Computer Applications. 2014; 43:121-41.

12. Shu J, Shen Z, Xue W. Shield: A stackable secure storage system for file sharing in public storage. Journal of Parallel and Distributed Computing. 2014; 74:2872-83.
13. Akshaya B, Sudha C, Suvedha B, Shanthi P, Umamakeswari A. Efficient ABBE for improving cloud security in a dynamically changing user environment. Indian Journal of Science and Technology. 2015 May; 8(9):306-11.
14. Rajarajeswari S, Somasundaram K. Data confidentiality and privacy in cloud computing. Indian Journal of Science and Technology. 2016 Jan; 9(4):1-8.
15. Kumari PS, Kamal ARNB. Optimal integrity policy for encrypted data in secure storage using cloud computing. Indian Journal of Science and Technology. 2016 Feb; 9(8).
16. Suveetha K, Manju T. Ensuring confidentiality of cloud data using homomorphism encryption. Indian Journal of Science and Technology. 2016 Feb; 9(8).
17. Monikandan S, Arockiam L. Confidentiality technique to enhance security of data in public cloud storage using data obfuscation. Indian Journal of Science and Technology. 2015 Sep; 8(24):88-97.
18. Brindha T, Shaji RS, Rajesh GP. A survey on the architectures of data security in cloud storage infrastructure. International Journal of Engineering and Technology. 2013 Apr-May; 5(2):1108-14.
19. Dia S, Wang C, Chen L. Ex-post efficient resource allocation for self-organizing cloud. Computers and Technical Engineering. 2012 Jul 2:1-32.
20. Di S, Robert Y, Vivien F, Kondo D, Wang C, Cappello F. Optimization of cloud task processing with checkpoint-restart mechanism. ACM. 2013 Nov:17-21.
21. Karimi M. Hybrid discrete particle swarm optimization for task scheduling in grid computing. International Journal of Grid Distribution Computing. 2014; 7(4):93-104.
22. Khedkar SV, Gawande AD. Data partitioning technique to improve cloud data storage security. International Journal of Computer Science and Information Technologies. 2014; 5(3):3347-50.
23. Jadhav SP, Nandwalkar BR. Efficient cloud computing with secure data storage using AES. International Journal of Advanced Research in Computer and Communication Engineering. 2015 Jun; 4(6):377-81.