Volume 4, No. 8, May-June 2013

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Secure Data Forwarding and Dependable Storage Services in Hybrid Cloud Computing

Lishabeni W Lotha
Department of Computer Science & Engineering
East West Institute of Technology
Bangalore, India
lishabeni@yahoo.com

Jyothi R
Department of Computer Science & Engineering
East West Institute of Technology
Bangalore, India
jyothirewit@gmail.com

***Abstract*—** For an increased level of scalability, availability and durability, some customers may want their data to be replicated on multiple cloud servers. The more copies the cloud service provider (CSP) is asked to store, the more fees the customers are charged. Moreover, it allows authorized users (i.e., those who have the right to access the owner's file) to seamlessly access the file copies stored by the CSP, and supports public verifiability. The proposed scheme is proved to be secure against colluding servers. However, the importance of cloud server also makes them attractive to attackers. In this paper, to provide confidential and reliable data transfer from the data owner to the remote user through the cloud server, we construct an efficient provable data possession scheme for distributed cloud storage. Here confidentiality or privacy is protected by, making the request to the data owner. If the requested user is registered in the data owner then the data owner will be providing the secret key to the requested user. Using that secret key the user will request the cloud server for a particular file. If the secret key matches in the cloud server, then the user will receive the file. Else the user will not receive the file and he will be detected as an attacker. Integrity is provided by, giving the data owner the capability to update any modification done to the data stored in the cloud server.

Keywords- Confidentiality; Reliability; Encryption; Network Efficiency; Cloud Server.

I. INTRODUCTION

Cloud Computing environment is a heterogeneous environment combining thousands to millions of cloud servers. We call such a distributed cloud environment as multi-cloud or hybrid cloud [1]. Cloud computing have been widely deployed for various applications like data storage, environmental data collection, medical science, IT services etc.However, Security becomes extremely important factor when cloud servers are randomly deployed in a hostile environment. If such an important platform is vulnerable to security attacks, it would bring irretrievable losses to the clients. For example, the confidential data in an enterprise may be illegally accessed through a remote interface provided by a multi-cloud, or relevant data and archives may be lost or tampered with when they are storedinto anuncertain storage pool outside the enterprise. Therefore, it is indispensable for cloud service providers (CSPs) to provide security techniques for managing their storage services. In this papera data storage service involves three different entities: Clients who have a large amount of data to be stored in multiple clouds and have the permissions to access and manipulate stored data; Cloud Service Providers (CSPs) who work together to provide data storage services and have enough storages and computation resources; and Trusted Third Party (TTP) who is trusted to store verification parameters and offer public query services for these parameters [7].

In this structure, the data owner sends the data into the cloud server by dividing the data into n number of blocks. Data owner can verify if any data block in cloud server is modified. Remote user has to provide its authorization to access the data from cloud server by providing the secret key provided by the data owner. As an intermediate tier, cloud servers are responsible for hosting raw data and replying queries.

The TTP servesas an intermediate tier between the data owner and the remote user for storing data and processing queries. A TTP server is constructed as a core trust base on the cloud for the sake of security. We assume the TTP is reliable and independent through the following functions: to setup and maintain the CPDP cryptosystem; to generate and store data owner's public key; and to store the public parameters used to execute the verification protocol in the CPDP scheme. The TTP is not directly involved in the CPDP scheme in order to reduce the complexity of cryptosystem. Trusted Third Party (TTP) whois trusted to store verification parameters and offerpublic query services for these parameters [7]. In our system the Trusted Third Party, view the user data blocks and uploads to the distributed cloud. In distributed cloud environment each cloud has user data blocks. If any modification is made by cloud owner an alert is send to the Trusted Third Party. The inclusion of cloud server also brings significant security challenges. As cloud server stores data received from data owner and remote user, it acts as an important role for answering queries; they are more vulnerable to be compromised, especially in a hostile environment. A compromised cloud server imposes significant threats to a cloud computing environment. First, the attacker may obtain sensitive data that has been, or will be, stored in the cloud server. Second, the compromised cloud server may return forged data for a query. Third, this cloud server may not include all data items that satisfy the query.

Therefore, we propose a scheme that prevents attackers from gaining information from cloud server collected data and allows the TTP to detect compromised cloud server when they misbehave. For confidentiality, the data owner does not allow

unauthorized users to access information. For reliability, if the data is modified, remote user is not provided the data until it is updated by the cloud server. The main contributions of this paper include:

a. We provide a novel mechanism for secure data storage in the encryption domain.
b. We present an efficient data structure to guarantee the integrity of data [8].

II. RELATED WORK

A. *Ensuring Data Storage Security in cloud computing:*

Cloud Computing has been envisioned as the next generation architecture of IT Enterprise. In contrast to traditional solutions, where the IT services are under proper physical, logical and personnel controls, Cloud Computing moves the application software and databases to the large data centers, where the management of the data and services may not be fully trustworthy. This unique attribute, however, poses many new security challenges which have not been well understood. In this article, we focus on cloud data storage security, which has always been an important aspect of quality of service. To ensure the correctness of users' data in the cloud, we propose an effective and flexible distributed scheme with two salient features, opposing to its predecessors. By utilizing the holomorphic token with distributed verification of erasure-coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., the identification of misbehaving server(s). Unlike most prior works, the new scheme further supports secure and efficient dynamic operations on data blocks, including: data update, delete and append. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

B. *Provable Data Possession at Untrusted Stores:*

We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it [2]. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems [3] , [4]. We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees [3], [4]. In particular, the overhead at the server is low (or even constant), as op-posed to linear in the size of the data. Experiments using our implementation verify the practicality of PDP and re- veal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.

III. MODELS AND PROBLEM STATEMENT

A. *System Model:*

A multi cloud computing consists of three types of entities as shown in Fig. 1.Clients who have a large amount of data to be stored in multiple clouds and have the permissions to access and manipulate stored data; Cloud Service Providers (CSPs) who work together to provide data storage services and have enough storages and computation resources; Trusted Third Party (TTP) who is trusted to store verification parameters and offer public query services for these parameters; and a Remote user who can access data from cloud server provided he has correct Secret key and file name.

B. *Threat Model:*

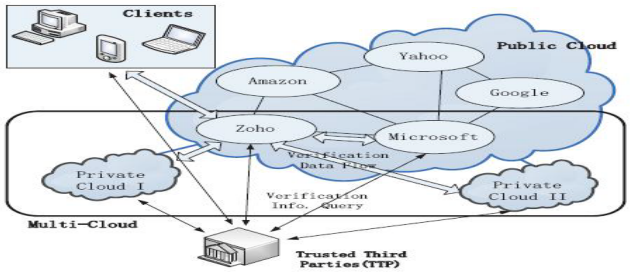
For a cloud computing environment, we neither assume that CSP is trust to guarantee the security of the stored data, nor assume that data owner has the ability to collect the evidence of the CSP's fault after errors have been found. If a cloud server is compromised, the subsequent collected data of the data owner will be known to the attacker, and the compromised cloud server may send forged data. After a cloud server is compromised, the large quantity of data stored on the server will be known to the attacker, and upon receiving a query from the remote user, the compromised cloud server may return a falsified result formed by including forged data or excluding legitimate data.

A TTP server is constructed as a core trust base on the cloud for the sake of security. We assume the TTP is reliable and independent through the following functions: to setup and maintain the CPDP cryptosystem; to generate and store data owner's public key; and to store the public parameters used to execute the verification protocol in the CPDP scheme.

C. *Problem Statement:*

The fundamental problem for a multi cloud computing environment is the following: How can we design the cloud server scheme and the query protocol in a confidential and reliable manner. A satisfactory solution to this problem must meet the following two requirements.

a. Data confidentiality: The actual data stored in cloud server can be viewed only by an authorized user. If an authorized user tries to view the data, data owner can detect the misbehavior of cloud server and unauthorized user is able to view irrelevant data.
b. Data reliability: If a query result that a cloud server sends to the remote user either includes forged data or exclude



Verification architecture for data integrity.

Figure. 1. Architecture of multiple cloud servers

Legitimate data, the query result is not available to the remote user until it is updated by the data owner.

As shown in the architecture above, we have hybrid cloud computing environment, which includes both private and public cloud server [1].

IV. MODULES DESCRIPTION

A. Multi Cloud Storage Module:

Distributed computing is used to refer to any large collaboration in which many individual personal computer owners allow some of their computer's processing time to be put at the service of a large problem. In our system the each cloud admin consist of data blocks. The cloud user uploads the data into multi cloud. Cloud computing environment is constructed based on open architectures and interfaces; it has the capability to incorporate multiple internal and/or external cloud services together to provide high interoperability. We call such a distributed cloud environment as a multi-Cloud [1].A multi-cloud allows clients to easily access his/her resources remotely through interfaces.

B. Data Owner:

Data owner browses for the file to be uploaded in the cloud server, Send the data file to cloud server then splits file into 5 blocks. After which it generates HMAC code for each block then again generate secret key. It verify for any changes for the data stored in cloud server, if data is modified it updates data block or deletes the file

C. Remote User:

The remote user requests for the secret key from the cloud server, gains the permissions to access the stored data by providing valid received secret key .Then it enters the name of the he wants to receive from the cloud server.

D. Third Party Auditor:

Trusted third party who is trusted to store verification parameters and offer public query services for these parameters. In our system the Trusted Third Party, view the user data blocks and uploaded to the distributed cloud. In distributed cloud environment each cloud has user data blocks. If any modification tried by cloud owner an alert is send to the Trusted Third Party.

V. CONFIDENTIAL AND RELIABLE DATA STORAGE

In this section, we propose our scheme to preserve data confidentiality and provide reliability.

A. Confidential Storage:

The data is sent to cloud server by splitting it into five blocks. By splitting data into five blocks, network bandwidth and network overhead is reduced, efficiency is increased and the chance of the attacker to understand the actual data even when the storage node is compromised is reduced.Fig. 2 illustrates the steps to provide confidentiality.

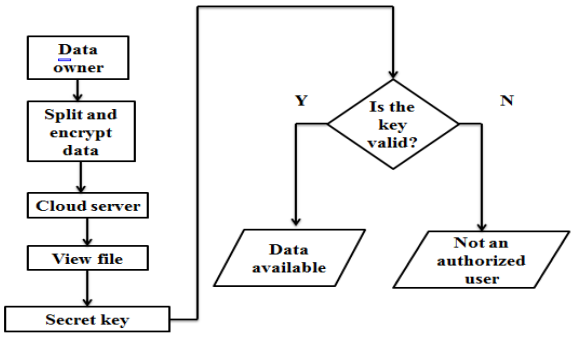


Figure.2. Flowchart to show confidentiality

Algorithm to provide confidentiality:

a. Split the data into n data blocks $d_1, d_2, \ldots, d_n$ where n=5
b. Encrypt the data using secret key $E(SK,d_i)$ where i=5
c. Let each of the n data block contain m bytes of data $b_1, b_2, \ldots, b_m$ where m=1024
d. For each data block sent, generate a hash code using SHA1, MD5

The data sent to cloud server contains the following information:

Data owner ⟶ cloud server: C_{id}, SK, $\{d_1, d_2, \ldots, d_5\}$, HMAC $\{ d_1, d_2, \ldots, d_5\}$

Where, Cid is the cloud server ID, SK is the Secret Key.

Confidentiality in cloud server is ensured by providing access only to authorized users through the secret key shared between data owner and cloud server. Here every cloud server and the data owner share a secret key for a time interval TI, which makes up a one-way key chain i.e., let SK_tbe the secret key of cloud server Ci at time interval t, SK_t = hash (SK_t, t−1). After the time interval, the old key is erased from the sensor and a new key is generated by the embedded hash function. Thus compromising a cloud server at time t, does not lead to disclosure of the information stored at time t-1. After remote user receives the query reply from cloud server, the received data is decrypted using the shared key between the corresponding cloud server and the remote user.

B. Reliability:

Reliability ensures that remote user gets the exact reply for the query issued to cloud server.If the data is modified, the remote user will not get the actual data until it is updated by cloud server.

Algorithm to provide reliability:

a. The HMAC values of the data blocks sent is stored in the cloud server.
b. The cloud server can verify for integrity of each data block it has sent.
c. Compare the HMAC value of the data block in cloud server with HMAC value of data block in remote user.
d. If the values are same there is no modification in data; if values are different, it indicates data modification.

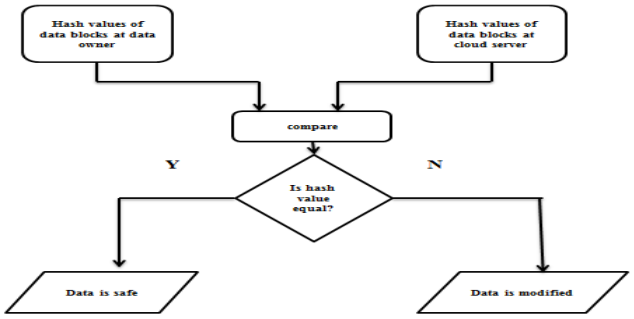


Figure.3. Flowchart to show reliability

A forged message is detected if a parent's calculated HMAC is inconsistent with HMAC received from the child. Fig. 3 illustrates the steps to provide reliability.

Verification of block d_1 for eg:

If HMAC (d_1) at sensor = HMAC (d_1) at storage node => No data modification.

If HMAC (d_1) at cloud server! =HMAC (d_1) at storage node => Data modification.

VI. CONCLUSION

The benefits and problems of the cloud computing in hybrid cloud computing are discussed. Confidential communication is achieved by making the cloud server to provide access to authorized users only. Data Integrity is provided by, giving the data owner the capability to update any modifications done to the data stored in cloud server [8]. For future work ,One more addition can be done for more security i.e. by sending a mobile alert to the data owner if a hacker/attacker tries to update/modify the data stored in the cloud server, In this way, security in hybrid cloud computing is much more strengthened.

VII. REFERENCES

[1] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. T. Foster, "Virtual infrastructure management in private and hybridclouds," IEEE Internet Computing, vol. 13, no. 5, pp. 14–22, 2009..

[2] G. Ateniese, R. C. Burns, R. Curtmola, J. Herring, L. Kissner, Z. N. J. Peterson, and D. X. Song, "Provable data possession at untrusted stores," in ACM Conference on Computer andCommunicationsSecurity, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 598–609.

[3] A. Juels and B. S. K. Jr., "Pors: proofs of retrievability for large files," in ACMConference on Computer and CommunicationsSecurity, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 584–597.

[4] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedingsof the 4th international conference on Security and privacy incommunication netowrks, SecureComm, 2008, pp. 1 10.

[5] C. C. Erway, A. K¨upc¸ ¨u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in ACM Conference onComputer and Communications Security, E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds. ACM, 2009, pp. 213–222

[6] H. Shacham and B. Waters, "Compact proofs of retrievability," in *ASIACRYPT*, ser. Lecture Notes in Computer Science, J. Pieprzyk, Ed., vol. 5350. Springer, 2008, pp. 90–107.

[7] Q. Wang, C.Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *ESORICS*, ser. Lecture Notes in Computer Science, M. Backes and P. Ning, Eds., vol. 5789. Springer, 2009, pp. 355–370.

[8] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau, "Dynamic audit services for integrity verification of outsourced storages in clouds," in *SAC*, W. C. Chu, W. E. Wong, M. J. Palakal, and C.-C. Hung, Eds. ACM, 2011, pp. 1550–1557.