Survey on Secure Protocol for Data Storage in Cloud Computing

Preethi
8th semester, Dept. of Information Science and Engineering
JSSATE
Bangalore, India
indian.preethi@gmail.com

Rekha P. M.
Assistant Professor,
Department of Information Science and Engineering
JSSATE
Bangalore, India
rekhapm12@gmail.com

***Abstract*— Cloud computing is an Internet based computing model, which supports on demand access to the shared pool of configurable computing resources. User's data is stored in remotely located large data centers. The clients can get access and modify the stored data over the Internet. The Third Party Auditor (TPA) monitors the data on behalf of the client. Therefore the data stored on the servers lack integrity. Cloud services should ensure data integrity and provide trust to the users' privacy. Due to this outsourcing of data, integrity and security of data becomes a tough challenge. In this paper, we present a survey of data integrity auditing mechanism in cloud storage, and provide a detailed comparison of integrity checking protocols.**

Keywords—Cloud computing, data integrity, cloud security, Third Party Auditor (TPA).

I. INTRODUCTION

Remote storage of data over the network is flexible because of increase in the speed and bandwidth of the Internet. The network architecture for cloud storage service is shown in Fig. 1. Three different network entities are:

- User: an entity, who relies on cloud for data storage and computation, can be either enterprise or individual customer.
- Cloud Server (CS): an entity, which provides data storage service and has significant storage space and resources. CS is managed by cloud service provider (CSP).
- Third Party Auditor (TPA): an optional entity, trusted to access and expose risk of cloud storage service on behalf of the users upon request.

Moving huge data into cloud offers great convenience to users since it reduces burden of hardware management and data maintenance at local machines. Cloud computing provides various types of services such as data as a service, platform as a service and infrastructure as a service. It also offers a scalable, secure, reliable environment for clients at low cost. Although the cloud is powerful and reliable than the personal computing system there exists both internal and external threats for data integrity.

The user may not have a local copy of outsourced data, so the CSP may behave unfaithfully [3] with cloud users. For example: The CSP could discard the data which has not been

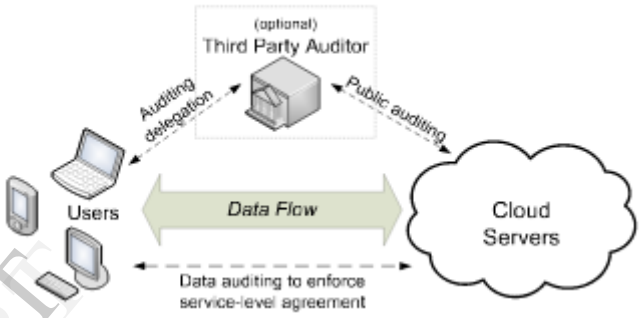

Fig. 1. Network architecture for cloud storage service.

accessed or rarely accessed to save the storage space. They may hide data losses to maintain the reputation of the organization. To overcome all these security and integrity threats CSP needs a mechanism to ensure the users' data integrity. Users can check the integrity of their data by themselves but it is inappropriate because neither CSP nor users could be guaranteed to provide unbiased auditing result. To resolve this problem the concept of third party auditing emerged. The TPA can do auditing work to convince both CSP and users

Traditional cryptographic primitives for data security protection can not be adopted [2] because users lose the control of data under cloud computing since verification of correctness of data must be done without explicit knowledge of the entire data. In this survey paper we compare different existing protocols for data integrity checking.

Section I gives the introduction to the paper. Section II reviews the related work. Section III explains various data integrity checking protocols. Section IV states the advantages and disadvantages of protocols and Section V provides the conclusion.

II. RELATED WORK

Joshi et al[4]in 2010 focused on different data security issues present in cloud computing and cloud environment. There are various benefits for enterprises by migrating to cloud and all these scopes are explained by sabahi [5]. His work also focuses on issues of security. Agarwal et al [6] in 2011 discussed some serious security threats in cloud computing. In 2012 M. Venkatesh et al [8] explained mechanism for

encrypting large data files and storage management which is based on RSA algorithm. Remagad et al [9] proposed an architecture which combines digital signature algorithm and Diffie Hellman and AES encryption.

Ateniese et al [10] is first to propose Provable Data Possession (PDP), his work explains how integrity of data can be checked which is stored at untrusted server without retrieving the entire file. But it only offers every user a limited number of verification request. Ateniese et al do not consider the case of dynamic data storage, if the proposed system is directly extended from static data storage to dynamic case it may suffer design and security problems.

III. PROTOCOLS FOR DATA INTEGRITY CHECKING

A. Hash Based Security

The hash value is calculated for each file before storing it into the server. To check the integrity this method will construct binary tree for all stored files (Fn) [1] based on the file size.

If the user wants to check the integrity of any file he will request TPA, then TPA will download those files completely and generate threads for each downloaded file to check the integrity simultaneously. The TPA will generate hash value of downloaded files and these hash values are compared with clients value if both matches then data integrity exist else the file is corrupted.

Example: If the client wants to check the integrity of F2 and F4 the TPA will download entire F2 and F4 and generate thread T2 and T4 and generate hash value for each thread. This value is sent to client, he will compare the hash values with his pre-computed hash value if matches data integrity is maintained by the server.

B. Two Way Handshake Protocol

Two-way handshake protocol relies on erasure correcting code for file distribution to provide both redundancies and data dependability. This code will redundantly disperse the data file F across n servers, where n=m+k; as shown in Figure 2: where m- data vectors; k- redundancy parity vector. “k” is chosen from m in such a way that original data vector can be retrieved from any m out of m+k data- parity vector. The original data file i.e. F can survive the failure any k out of m+k servers.

Before distributing file F generate certain number of tokens from data vector m. These tokens will cover subset of data blocks. As illustrated in Fig. 3 when user wants to check the integrity of their data he can challenge the CS by randomly generating set of block indices on receiving this challenge, server will generate "signature"[2] of specified block indices and sends back to user upon receiving user can compare signature and pre-computed token if they match the data is secure.

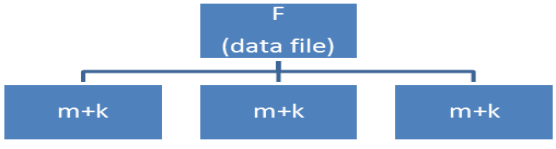

Fig. 2. Redundant File Storage

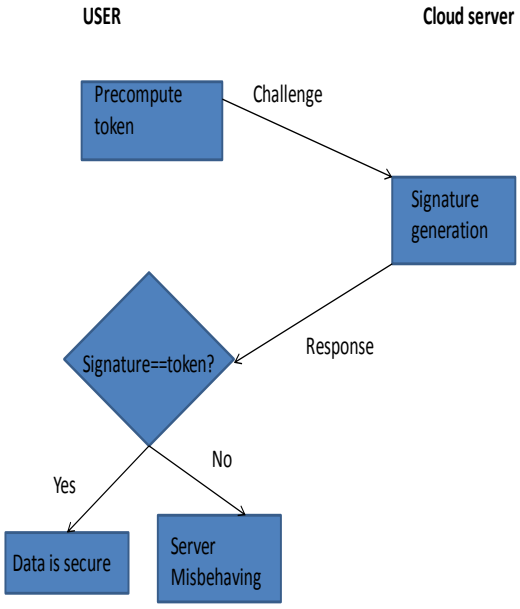

Fig. 3. Two-way Handshake protocol

C. PBA Protocol

PBA protocol works in two stages:

1. Setup Phase
2. Batch Verification Phase

During setup stage user will generate key and signature using keygen and signgen algorithm and passes to the TPA.TPA will upload the data and signature to the organizer and deletes its local copy. Organizer will disperse data into different Cloud server security (CSS) and generate index table to note which data block is stored in which CSS.

The batch verification is done by TPA to check the integrity of outsourced data. It has 5 Algorithm [3]. The sequence diagram is as illustrated in Figure 4. The sequence is as follows:

1. Challenge
2. Forward
3. Prove
4. Response
5. Batch verify

1. TPA will send set of random index-coefficient pairs (Q) to the organizer as a challenge.
2. Then organizer will forward this challenge to the particular CSS based on index table.
3. Each CSS which received the challenge will return n aggregated proofs to the organizer.
4. After recieving proofs the organizer will encode the proofs and generate fianl response and send it to TPA.
5. The TPA will decode this response and verifies the response in a batch way.

If the batch auditing pass the data in integrated or else data is corrupted.

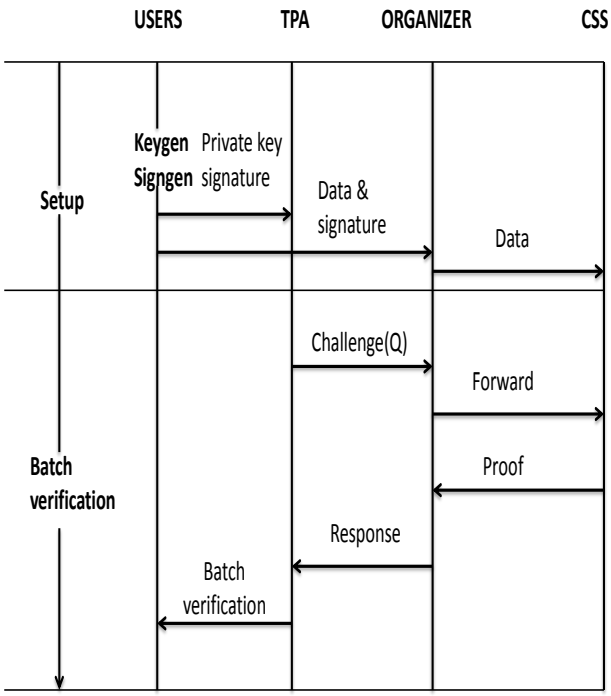

Fig. 4. Sequence diagram for PBA Protocol

D. Diffie Hellmann Key Exchange Protocol

Diffie Hellmann Key Exchange protocol is proposed by witfield diffie and Martin Hellman in 1976.It is first public key cryptography Scheme. It uses two keys - private and public key. Sender will encrypt the message by his private key and receiver public key. Receiver will decrypt the message by his private key and sender's public key.

Once the user account is created at cloud server the connection between user and CSP is established by Diffie Hellmann Key Exchange (DHKE) protocol. The server will generate unique user id for each of its user, the Diffie Hellmann equivalent stream, private and public key. The UID is sent to user which acts as a tool of authentication each time he logs in.

When the user logs in, the cloud server will authenticate by checking Diffie Hellmann equivalent of UID from the server repository. If the key matches only then user can access cloud.

For data exchange as in Figure 5 client will encrypt his query using public key and sent to server. The server will decrypt the query using private key process the query, the result of query is encrypted again and sent back to the client.

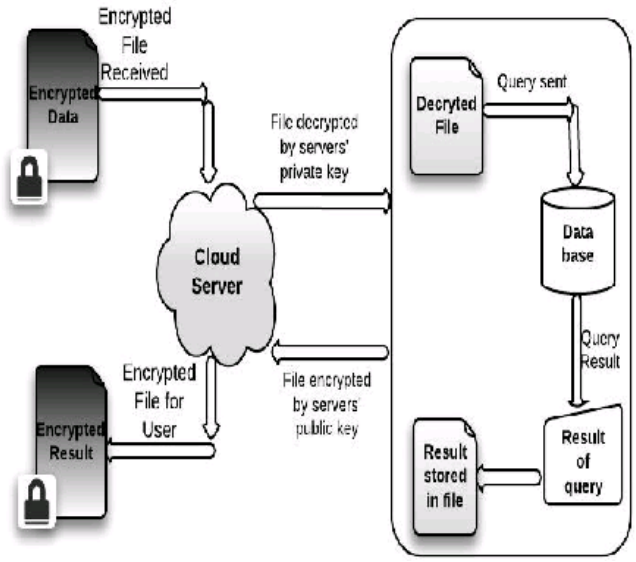

Fig. 5. Data exchange using DH key

IV. COMPARISON

In hash based algorithm integrity is checked by TPA, which reduces client's burden. But to check the integrity of the data we need to download entire file which increases I/O cost and storage cost. If Two Way Handshake protocol is compared with traditional replication based file distribution technology, there is significant reduction in communication and storage overhead. But number of challenges by user to server is limited by number of precomputed verification tokens. The PBA protocol will reduce total auditing time and communication cost in multi-cloud storage(if performed in group).if individual auditing is applied in multi-cloud, it will cause significant communication and computation cost. Diffie Hellman Key exchange protocol is significantly better for establishment of connections between user and cloud server. But for single conversation between two parties requires four keys.

V. CONCLUSION AND FUTURE WORK

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. In this survey paper, we study about issues related with cloud storage and security, the different existing integrity checking protocols for data stored in cloud.

As our future work we focus on reducing the impact in maintaining the challenge key in user's local space. For this we can split the challenge key into several parts- partial keys and maintain those keys in different cloud server and yet ensure security and data transparency. This might reduce the space overhead and possible cross verification of the verification process of a TPA by other TPA's.

ACKNOWLEDGMENT

The authors thank Dr. D. V. Ashoka, professor and Head of Department of Information Science and Engineering, JSS Academy of Technical Education, Bangalore for their constant review and support in writing this paper.

REFERENCES

[1] V.Tejaswini,S.k.Prashanth,Dr.N. sambasiva Rao, C. Satya kumar.Privacy and Integrity Preserving in Cloud Storage Devices.IOSR Journal of Computer Engineering(IOSR-JCE)Volume 12,Issue 5(Jul-Aug.2013)

[2] M.R.Tribhuwan,V.A.Bhuyar,Shabana Pirzade.Ensuring Data Storage Security in Cloud Computing through Two-way Handshake based on Token Management.IEEE Computer Society 2010.

[3] He Kai,Huang Chuanhe,Wang Jinhai,Zhou Hao,Chen Xi,Lu Yilong,Zhang Lianzhen,Wang Bin.IEEE 2013 8th Annual ChinaGrid Conference.

[4] Joshi, J.B.D., Gail-Joon Ahn. Security and Privacy Challenges in Cloud Computing Environments. IEEE Security Privacy Magazine, Vol 8, IEEE Computer Society, 2010, p.24-31.

[5] Farzad Sabahi. Cloud Computing Security Threats and Responses. Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference.

[6] Ashish Agarwal, Aparna Agarwal. The Security Risks Associated with Cloud Computing. International Journal of Computer Applications in Engineering Sciences [VOL I, SPECIAL ISSUE ON CNS, JULY 2011] [ISSN: 2231-4946].

[7] Ashutosh Kumar Dubey, Animesh Kumar Dubey, Mayank Namdev, Shiv Shakti Shrivastava. Cloud-User Security Based on RSA and MD5 Algorithm for Resource Attestation and Sharing in Java Environment. Software Engineering (CONSEG), CSI Sixth International Conference, Sept. 2012.

[8] M.Venkatesh, M.R.Sumalatha, Mr.C.SelvaKumar. Improving Public Auditability, Data Possession in Data Storage Security for Cloud Computing. Recent Trends In Information Technology (ICRTIT), 2012 International Conference, April 2012.s

[9] Prashant rewagad,yogita Pawar in. Use of digital signature with Diffie Hellman key exchange and AES encryption algorithm to enhance data security in cloud computing.2013 International Conference On Communication system s and network Technologies.

[10] G.Ateniese,R. Burns,R. Curtmola,J. Herring,L. Kissner,Z. Peterson,and D. Song,"Provable Data Possession at Untrusted Stores,"in Proc.ACM CCS,2007, pp.598-610.

[11] T. Velte, A. Velte, and R. Elsenpeter, Cloud Computing: A Practical Approach, first ed., ch. 7. McGraw-Hill, 2010.

[12] A. Juels and B.S. Kaliski Jr., "PORs: Proofs of Retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 584-597, Oct. 2007.

[13] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), pp. 598-609, Oct. 2007.

[14] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, "Auditing to Keep Online Storage Services Honest," Proc. 11th USENIX Workshop Hot Topics in Operating Systems (HotOS '07), pp. 1-6,2007.

[15] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song, "Provable Data Possession at UntrustedStores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), pp. 598-609, 2007.